# A Biological Control Model of a One-Link Neuromusculoskeletal Sagittal Arm

A Thesis

Presented in Partial Fulfillment of the Requirements for

the Degree of Bachelor of Science with Honors

at The Ohio State University

By:

Michael Lucas

The Ohio State University

March 2006

Honors Examination Committee:

Dr. Hooshang Hemami

Dr. Andrea Serrani

Approved by:

_____

Advisor

Electrical and Computer Engineering

**Abstract**

The focus of this thesis is the analysis of stability, point-to-point, and periodic motion of a one link neuromusculoskeletal sagittal arm with and without linear tendons. The stability of equilibria is examined for models with and without tendons in an effort to deduce the role the aforementioned organ plays in the overall system dynamics. Simple cases of periodic and point-to-point movements are also analyzed. The tracking of a reference input is formulated as an sub-optimal control problem with control laws derived from system linearizations. Optimality criteria are incorporated into the control law design process via minimization of cost functions based on the arms end point variance. We hope to determine the optimal configurations necessary for biological efficiency and to provide insight into the biophysical rules behind such strategies. Additionally, potential causes of deficient movement control, such as those present in Parkinson's disease, are modeled by carefully selected simulations of periodic movements. Such simulations may provide clues to future courses of treatment.

**Acknowledgments**

I would like to express my sincere thanks to all of those who helped me throughout the course of this project. First and foremost, I would like to thank my advisor Professor Hooshang Hemami. Without his sage advice and heartfelt encouragement this project would not have been completed. I also send my sincere thanks to Professor Andrea Serrani for always giving me access to his expertise at a moment's notice.

I would be remiss if I failed to mention my parents. I would never have survived my undergraduate years without their love and support. Indispensable to the completion of this project were the good times I shared with: Matt Charnas, Troy Judy, Chad Lester, Zachary Parkins, Sis and lil Sis, Chris Smith, and many others. I am also indebted to Nathan Bloom and Mark Elias who kept me company on my innumerable all nighter's in DL 517, and shared my daily exuberance for southwest chicken submarine sandwiches.

Finally, I dedicate this project to Beth. I would like to thank her for putting up with me spending most of my time at the lab. She provided an escape from the demands and constraints of engineering work, and supported me in more ways than she will ever know.

# Contents

**1 Introduction** **7**

**2 The System Model** **10**

2.1 A Second Order Model . . . . . 10

2.1.1 Uncontrolled System Analysis . . . . . 11

2.1.2 Derivation of the Flexor and Extensor Actuators . . . . . 12

2.1.3 The Controlled System . . . . . 16

2.2 A Sixth Order Model: Muscle-like Actuators with Tendons . . . . . 18

2.2.1 The Complete Sixth Order System . . . . . 22

2.3 Summary . . . . . 25

**3 Stability Properties and Point-to-Point Movement of the Sagittal Arm** **26**

3.1 Stability of the Second Order Model . . . . . 27

3.2 Stability of the Sixth Order Model . . . . . 29

3.3 Stable Point-to-Point Movements . . . . . 30

3.3.1 Point-to-Point Movements of the Second Order System . . . . . 31

3.3.2 Point-to-Point Movements of the Sixth Order System . . . . . 32

3.4 Summary . . . . . 38

**4 Formulation of a Control Problem for Complex Motion** **40**

4.1 A Discrete Time Sub-Optimal Control Problem . . . . . 42

4.2 Linearization of the Second Order Model . . . . . 43

4.3 The Discretized System . . . . . 45
4.4 Sub-Optimal Control of the Discretized Linearized Second Order Model . . . 46
4.4.1 Asymptotic Tracking Control . . . . . 48
4.5 Feedback Linearization of the Second Order Model . . . . . 53
4.6 Summary . . . . . 54

**5 Discussion and Conclusions** **56**

**A The Hartman-Grobman Theorem** **58**

**B Differentiation of the Heaviside Step Function** **59**

**C Jacobian Linearization of the Second Order Model** **63**

**D Matlab Functions** **66**

# List of Figures


1.1 The Sagittal Arm . . . . . 8

2.1 The Second Order Model . . . . . 17

2.2 The Sixth Order Model . . . . . 23

3.1 Second Order Model Equilibrium Stability, State Trajectories . . . . . 28

3.2 Second Order Model Equilibrium Stability, Phase Portrait . . . . . 28

3.3 Second Order Model Equilibrium Stability with Spindle Feedback, State Trajectories . . . . . 29

3.4 Point-to-Point Motion: Neural Firing Inputs $R_1$ and $R_2$ . . . . . 31

3.5 Point-to-Point Movement: Second Order Model without Spindle Feedback, State Trajectories . . . . . 32

3.6 Point-to-Point Movement: Second Order Model without Spindle Feedback, Phase Portrait . . . . . 33

3.7 Point-to-Point Movement: Second Order Model with Spindle Feedback, Phase Portrait . . . . . 33

3.8 Point-to-Point Movement: Second Order Model with Spindle Feedback, State Trajectories . . . . . 34

3.9 Point-to-Point Movement: Sixth Order Model with Spindle and Golgi Feedback, State Trajectories . . . . . 35

3.10 Point-to-Point Movement: Sixth Order Model with Spindle and Golgi Feedback, Phase Portrait . . . . . 35

3.11 Point-to-Point Movement: Sixth Order Model with Spindle and Golgi Feedback, Muscle Forces . . . . . 36
3.12 Point-to-Point Movement: Sixth Order Model with Spindle Feedback, State Trajectories . . . . . 36
3.13 Point-to-Point Movement: Sixth Order Model with Spindle Feedback, Phase Portrait . . . . . 37
3.14 Point-to-Point Movement: Sixth Order Model with Golgi Feedback, State Trajectories . . . . . 37
3.15 Point-to-Point Movement: Sixth Order Model with Golgi Feedback, Phase Portrait . . . . . 38

4.1 State and Reference Trajectories . . . . . 49
4.2 Muscle Forces . . . . . 50
4.3 Control Inputs . . . . . 51
4.4 Control Gain Matrix Elements . . . . . 52
4.5 The Feedback Linearized System . . . . . 54

B.1 Differentiating the Heaviside Step: A Graphical View . . . . . 60

# List of Tables

2.1 Parameters of the Second Order Model . . . . . 18
2.2 Parameters of the Sixth Order Model . . . . . 24

# Chapter 1

# Introduction

The system studied in this thesis is a one-link dynamical model of a Neuromusculoskeletal Arm hinged to a inertial reference frame. For simplicity, motion of the arm is restricted to the sagittal plane. The trajectory of the arm is defined by the angular position with respect to the vertical, $\theta$, and angular velocity, $\dot{\theta}$. The arm is acted upon by two muscle like actuators that loosely approximate the bicep and triceps. The bicep like actuator moves the arm in a counterclockwise direction, while the triceps actuator moves the arm clockwise. The actuators are controlled by neurological input signals related to neural firing rates.

The system is characterized by the following physical parameters: its mass, $m$; distance from the arm's center of gravity to the hinge, $d$; the arm's moment of inertia about the hinge, $J$; and the moment arms of the actuators, $k1$ and $k2$. In this thesis the moment arms are assumed equal and constant. A pictorial representation of the system is given below:

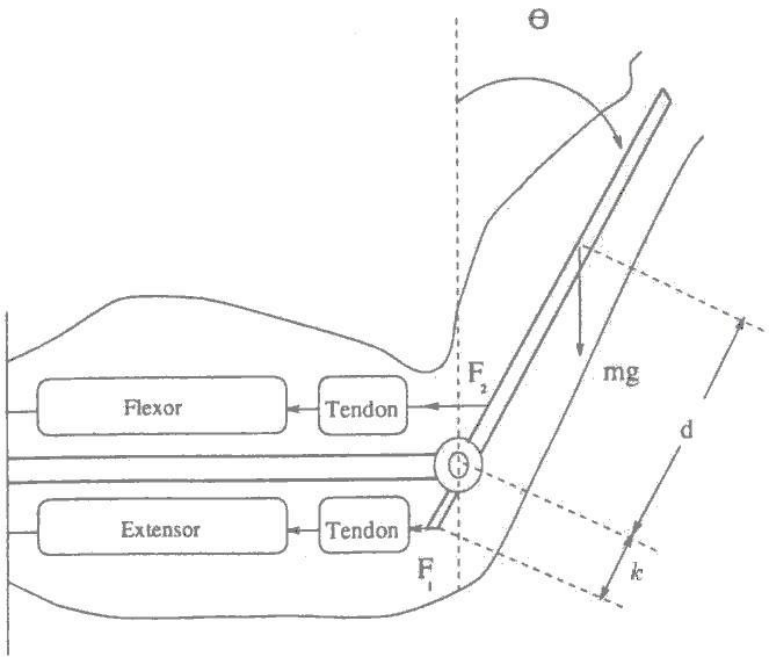


Figure 1.1: The Sagittal Arm [6]

Working with this simple second order model, Dinneen and Hemami showed that stable movement could be achieved by coactivating the actuators with sufficiently high neural firing rates. Successful point-to-point movements were then performed with and without spindle feedback. Additionally, the inclusion of the delays inherent in natural afferent and efferent signal pathways was shown to generate undesirable oscillations. This oscillatory behavior was likened to the movement of persons afflicted with Parkinson's disease [2].

Further research was done on a similar model by Kim and Hemami. In these experiments, the previous models were updated to include tendons by utilizing the work of F.E. Zajac, among others [6],[14],[15]. Zajac has published several important results on the dynamical properties of tendons, including their low pass filtering effects. Employing the newly created model, successful movement was demonstrated using burst signals as control inputs.

Despite these successes, much work remains unfinished. More results on the stability properties of models including tendons are necessary for research in this area to progress. These results will help determine what effects tendons have on controlling neural firing signals. Moreover, there are very little results available for any type of complex movement. The

extension of previous research to more complex movements is a logical step forward. Contributions to the qualitative understanding of how the brain controls the arm in continuous movement would prove invaluable for numerous applications. Thus, the two primary goals of this thesis are to expand upon the existing results for stability of models with tendons and to provide some preliminary data on complex motion of the sagittal arm.

In Chapter Two we present the two models that will be employed in experimentation, the second order model that has been previously described and a sixth order model that incorporates linear tendons and golgi feedback organs. These models are realized in the Simulink simulation environment, in which all experimentation was performed [1]. In Chapter Three we discuss some fundamental stability properties of the second and sixth order models, namely stability of equilibria and point-to-point movement. Chapter Four explores some control strategies of the arm under more complex movement. Finally, Chapter Five concludes the thesis with a summary of results and some possible directions for future research.

# Chapter 2

# The System Model

## 2.1 A Second Order Model

The sagittal arm, as represented in Figure (1.1), is governed by the following differential equation,

$$J\ddot{\theta} - mgd\sin\theta = k(F_1 - F_2) \tag{2.1}$$

where $k$, $F_1$, and $F_2$ denote the equal moment arms and applied forces of the tricep and bicep like actuators, respectively.

The system can be written in state space form by defining the state variables,

$$x_1 = \theta$$

$$x_2 = \dot{\theta}$$

and decomposing (2.1) into a pair of first order ODE's, given by:

$$f_1(x) = \dot{x_1} = x_2$$

$$f_2(x) = \dot{x_2} = \frac{mgd}{J}\sin x_1 + \frac{k}{J}(F_1 - F_2) \tag{2.2}$$

### 2.1.1 Uncontrolled System Analysis

It is worthwhile to examine the salient features of the homogeneous system model in order to understand the governing role played by neural control and to test the model's qualitative validity. Setting the derivative of the state vector ($\dot{x} = [\ \dot{x_1}\quad \dot{x_2}\ ]^T$) equal to zero and solving for equilibria (subject to $x_1 \in [0, \pi]$) yields $x = (0, 0)$ and $x = (\pi, 0)$. It is intuitively obvious that the equilibrium at the origin is unstable, as the system is similar to an inverted pendulum. This intuition can be confirmed with a straight forward application of the Hartman-Grobman Theorem, a complete formulation of which is given in Appendix A [5]. Linearizing the system about the origin yields

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{mgd}{J} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

The eigenvalues of the state matrix, $\lambda_1 = \sqrt{mgd/J}$ and $\lambda_2 = -\sqrt{mgd/J}$, reveal the origin is an unstable saddle point.

The stability of the second equilibrium point cannot be determined via linearization, and consequently requires slightly more sophisticated analysis. In a neighborhood around $x = (\pi, 0)$, $f_2(x)$ is well approximated by:

$$\dot{x_2} = \frac{mgd}{J}(\pi - x_1)$$

In order to use Lyapunov's second method, we shift the second equilibrium point to the origin by changing to the coordinates $\tilde{x_1} = x_1 - \pi$ and $\tilde{x_2} = x_2$. Choosing the Lyapunov function $V(\tilde{x}) = \tilde{x}^T P \tilde{x}$, where $P$ is the positive definite matrix defined as

$$P = \begin{bmatrix} \frac{mgd}{2J} & 0 \\ 0 & 1/2 \end{bmatrix}$$

The Lie derivative of $V(\tilde{x})$ with respect to the system is,

$$\dot{V}(\tilde{x}) = \frac{\partial V}{\partial \tilde{x}} f(\tilde{x}) = \frac{mgd}{J}\tilde{x}_1\tilde{x}_2 - \frac{mgd}{J}\tilde{x}_1\tilde{x}_2 = 0$$

Noticing that $\dot{V}$ is negative semidefinite for all $\tilde{x} \in \Re^2$ the equilibrium is stable. By invoking LaSalle's Invariance Principle it can be shown that the equilibrium exhibits attractivity, and therefore is asymptotically stable [4].

The model agrees well with our intuition of an uncontrolled limb. A normal force acting on the arm at rest in the first equilibrium position ($x = (0, 0)$) will cause it to fall. Such perturbations, even when small in magnitude and duration, cause the system to be overcome by gravitation. Similarly, an arm loose at one's side, as modeled by the second equilibrium, will return to its equilibrium state in the event of small disturbing forces.

### 2.1.2 Derivation of the Flexor and Extensor Actuators

From the homogeneous analysis it is obvious that the body must employ control to carry out stable point-to-point and continuous movements. Revisiting equation (2.2) the control is manifest in the extensor and flexor forces, $F_1$ and $F_2$. Note that throughout the remainder of the thesis the subscripts '1' and '2' refer to the extensor and flexor, respectively. In order to derive appropriate models of the flexor and extensor actuating forces the biophysical properties of muscle tissue, tendons (to be considered later), and feedback neural pathways must be investigated.

#### Muscle Model

The muscle model employed in this thesis assumes that the passive elasticity of an actuated muscle increases linearly with neural firing rate. The model also assumes that the muscle is composed of $N$ identical fibers. The tension in any individual fiber may then be modeled as a quadratic function of the deviation of the fiber from its equilibrium length, defined as $l$, and neural firing rate, $r$, given as:

$$v = (ar + cl)^2$$

Here $a$ is a proportionality constant that amplifies the neural firing rate and $c$ is the constant the defines the linear relationship between passive elasticity and firing rate. The total tension in the muscle is then:

$$V = Nv = N(ar + cl)^2 = (\sqrt{N}ar + \sqrt{N}cl)^2 = (\tilde{R} + Cl)^2 \tag{2.3}$$

where we have defined the parameters $\tilde{R} = \sqrt{N}ar$ and $C = \sqrt{N}c$ for notational convenience. According to Dinneen and Hemami, equation (3) accurately approximates the tension produced by fibers of identical length with different cross sectional areas [2]. Finally, a linear constant Hill effect is added to the model and the resulting muscular tension is

$$V = (\tilde{R} + Cl)^2 + B\dot{l}u(-\dot{l}) \tag{2.4}$$

Here $B$ is a viscous constant and $u(x)$ is the Heaviside step function defined as:

$$u(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{elsewhere} \end{cases}$$

Note that the muscular contraction velocity, $-\dot{l}$, and the viscous constant, $B$, are assumed sufficiently small to ensure positive tension. Equation (2.4) defines the muscle model where the flexor and extensor actuators are assumed identical.

**Control Signal Generation: Neural Feedback Mechanism**

The muscle like actuators are controlled by neural input signals derived from the desired trajectory and kinematic information fed back from spindle like sensors. The spindle like sensors approximate neural mechanisms that gather system information and transmit to the motor control centers (e.g. the brain and spinal cord). The sensor output is a signal that is a function of the muscle incremental length, $l$, and its velocity of movement, $\dot{l}$. The signal is nonzero only when two criteria are satisfied, the muscle is stretching ($\dot{l} > 0$) and the muscle length is greater than some threshold, $\gamma$. The sensor output $f$, according to Dinneen and Hemami can be approximated by [2]:

$$f = q\dot{l}^{\epsilon}(l - \gamma)u(l - \gamma)u(\dot{l}) \tag{2.5}$$

Where $q$ is a positive gain constant, $u(x)$ is the Heaviside step function, and $\epsilon$ is a real constant belonging to the interval $(0, 1)$. The flexor and extensor feedback signals are considered to be independent.

Within the spinal circuitry, each signal is amplified by the neural firing rate, $\tilde{R}$, and is then added to the rate. Thus, the neural control signals that comprise the input to the actuator models are given by:

$$R = \tilde{R}f + \tilde{R} = \tilde{R}q\dot{l}^{\epsilon}(l - \gamma)u(l - \gamma)u(\dot{l}) + \tilde{R} \tag{2.6}$$

Or in terms of the state variables:

$$\begin{aligned} R_1(x_1, x_2, u_1, u_2) &= \tilde{R_1}q(-kx_2)^{\epsilon}(l_o - kx_1 - \gamma)u(l_0 - kx_1 - \gamma)u(-kx_2) + \tilde{R_1} \\ R_2(x_1, x_2, u_1, u_2) &= \tilde{R_2}q(kx_2)^{\epsilon}(l_o + kx_1 - \gamma)u(l_0 + kx_1 - \gamma)u(kx_2) + \tilde{R_2} \end{aligned} \tag{2.7}$$

Thus, the control input to the system is given by,

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \tilde{R_1} \\ \tilde{R_2} \end{bmatrix} \tag{2.8}$$

It is important to note that each spindle's transmission path contains considerable delays, typically lasting around 40 ms. The total delay is the sum of the delay in the afferent path, $\Delta_a$, and the delay in the efferent path $\Delta_e$. These delays are modeled ideal and constant for all time. The flexor and extensor input signals are considered to be affected by identical delays.

### Actuator Dynamics

The model employs the state vector to derive the muscle length and muscle contraction velocity, $l$ and $\dot{l}$ respectively. Using the fact that the arm is hinged to an inertial reference frame, the lengths of the muscle-like actuators may be written as affine functions of the arm's angular position. The flexor and extensor lengths are then,

$$l_{fl} = l_0 + k\theta = l_0 + kx_1$$

$$l_{ex} = l_0 - k\theta = l_0 - kx_1 \tag{2.9}$$

Where $l_0$ is the initial muscle incremental length, assumed identical for each actuator, and $k$ is the moment arm of each actuator. In reality, $k$ is a function of both angular position and velocity and will vary slightly for each actuator. However, for analytical simplicity the moment arms of the flexor and the extensor are assumed constant and identical. The muscle contraction velocity may be found via differentiation with respect to time as,

$$\dot{l_{fl}} = +k\dot{\theta} = +kx_2$$

$$\dot{l_{ex}} = -k\dot{\theta} = -kx_2 \tag{2.10}$$

Thus, neural control signals (see equation (2.7)) may be generated from the state vector.

The force of each actuator can be found from equations 2.4, 2.9, and 2.10. The force of the extensor and flexor are given by:

$$F_1 = (R_1(x_1, x_2, u_1, u_2) + Cl_{ex})^2 + B\dot{l_{ex}}u(-\dot{l_{ex}}) = (R_1(x_1, x_2, u_1, u_2) - Ckx_1)^2 - Bkx_2u(kx_2)$$

$$F_2 = (R_2(x_1, x_2, u_1, u_2) + Cl_{fl})^2 + B\dot{l_{fl}}u(-\dot{l_{fl}}) = (R_2(x_1, x_2, u_1, u_2) + Ckx_1)^2 + Bkx_2u(-kx_2) \tag{2.11}$$

The model retains some properties of natural muscle. First, the actuator forces, $F_1$ and $F_2$, are always positive in isometric contractions where the contraction velocity is zero. Secondly, the actuator that is stretched applies a larger force than the one that is shortened. Each result is somewhat obvious and follows directly from the form of (2.11). Additionally, it is worthwhile to note that the addition of the Hill effect term (see section muscle) reduces the force of the respective actuator as the muscular contraction velocity increases.

### 2.1.3 The Controlled System

A complete description of the system can now be presented. Starting from the general model of the system given in equation (2.1), the controlled system model is derived. Substituting (2.11) into (2.1) yields,

$$J\ddot{\theta} - mgd\sin\theta = k(F_1 - F_2)$$

$$J\ddot{\theta} - mgd\sin\theta = k\Bigg(\Big((R_1 + Cl_{ex})^2 + B\dot{l_{ex}}u(-\dot{l_{ex}})\Big) - \Big((R_2 + Cl_{fl})^2 + B\dot{l_{fl}}u(-\dot{l_{fl}})\Big)\Bigg) \quad (2.12)$$

We suppress the state and control arguments of functions $R_1$ and $R_2$ for notational convenience. In state variable form the system is:

$$\dot{x_1} = x_2$$

$$\dot{x_2} = \frac{mgd}{J}\sin x_1 + \frac{k}{J}\Bigg((R_1 - Ckx_1)^2 - Bkx_2u(kx_2) - (R_2 + Ckx_1)^2 - Bkx_2u(-kx_2)\Bigg) \quad (2.13)$$

Equations 2.12 and C define the dynamics of the system. The second order model will be used throughout the remainder of this thesis to provide insight into the control problem that would otherwise be lost in the complexity of more realistic models. Additionally, the model is also useful to discern the effects of the linear tendons and golgi feedback that are introduced later. A system diagram is given below:

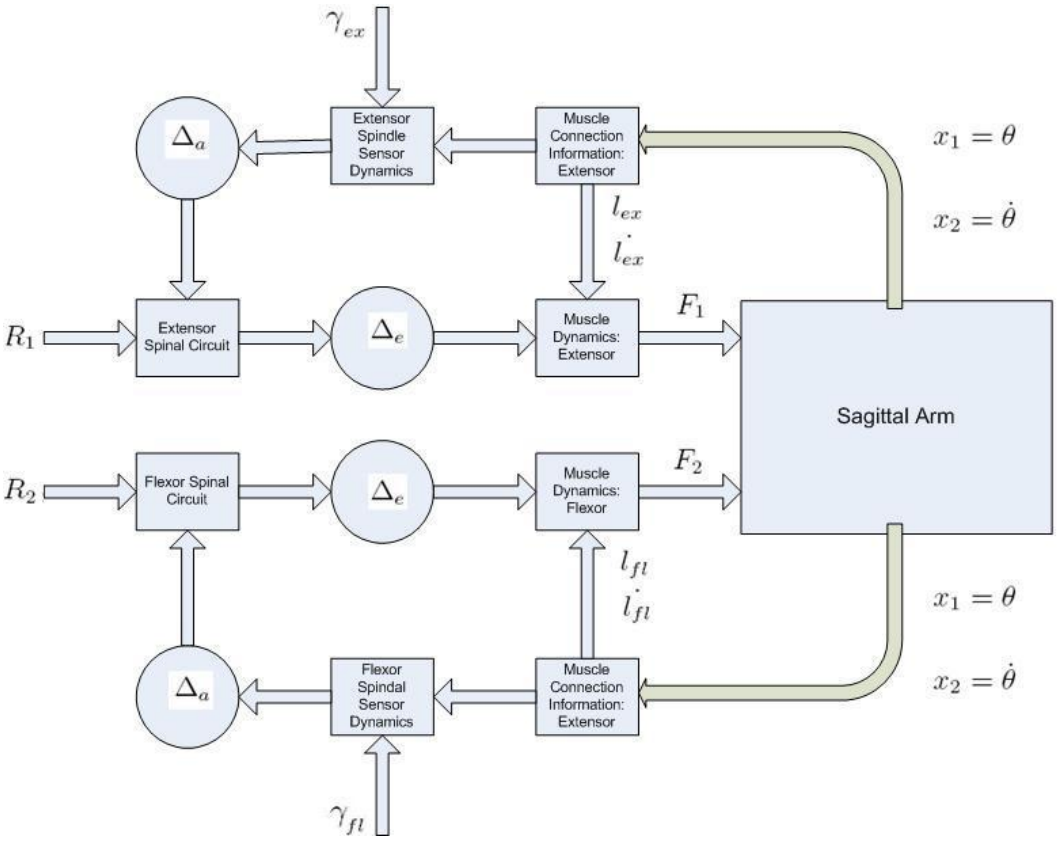


Figure 2.1: The Second Order Model has the inputs $\gamma_{ex}$, $\gamma_{fl}$, $R_1$, and $R_2$ which correspond the extensor and flexor threshold lengths and the neural firing inputs, respectively. The model has the angular position $x_1 = \theta$ and angular velocity $x_2 = \dot{\theta}$ as states. The output of the model is the system state. The afferent and efferent delays are denoted as $\Delta_a$ and $\Delta_e$, respectively.

A table of nominal values of system parameters that are used in simulation is given below:

| Parameter | Definition | Value |
|---|---|---|
| B | muscle Hill effect viscous constant | 75 |
| C | muscle passive elasticity constant | 30 |
| d | distance from arm's center of gravity to hinge | 0.18 [m] |
| g | gravity constant | 10 [m/$s^2$] |
| J | moment of inertia of arm about hinge | 0.075 [kg·$m^2$] |
| k | actuator moment arm | 0.04 [m] |
| $l_0$ | unstretched muscle fiber length | 0.3 [m] |
| m | mass of arm | 1.9 [kg] |
| q | spindle sensor gain | 2 |
| $\Delta_a$ | afferent delay | 0.035 [s] |
| $\Delta_e$ | efferent delay | 0.005 [s] |
| $\epsilon$ | spindle sensor contraction velocity exponent | 0.2 |
| $\gamma$ | muscle threshold length | 0 [m] |

Table 2.1: Parameters of the Second Order Model

## 2.2 A Sixth Order Model: Muscle-like Actuators with Tendons

Although the simple second order model can provide a great deal of insight into the dynamics of the system, it neglects key elements of the physical arm. In its biological form, the neuromusculoskeletal arm is a complex aggregate of muscles, tendons, ligaments, and neural pathways. In an attempt to more accurately model the intricacies of the arm, a slightly more complex model will also be utilized. Here linear tendons and golgi (force) feedback are incorporated in the model of each respective actuator. The control laws from this more complicated model will then be compared with those of the second order model to determine

the effects of tendons on the system dynamics. The derivation of the sixth order model starts from the complete second order model.

### Tendon Dynamics

The tendons are assumed identical for the two actuators. The output force of the muscle model comprises the input to the tendon model. The tendon dynamics are modeled using a Hookeian spring. Thus the tendon is governed by the equation:

$$F_m = k_t L_t u(L_t) = k_t(l_t - l_{t0})u(l_t - l_{t0}) \tag{2.14}$$

where $l_t$, $l_{t0}$, and $k_t$ are the tendon length, unstretched tendon length, and spring constant of the tendon, respectively, and $u(t)$ is the Heaviside step function. The Heaviside step function is included to ensure that the tendon model does not output any force when the organ is being compressed [15]. The tendon length can be computed at any time $t$ as $L_t = F_m/k_t$. However, because the tendon length is a dynamic quantity we define two additional states, $x_3 = L_{tex}$ and $x_4 = L_{tfl}$ to represent the extensor and flexor tendon lengths, respectively. Thus the tendon dynamics are completely described by the following state space model:

$$\begin{aligned} \frac{d(l_t - l_{t0})}{dt} &= \frac{d}{dt}\frac{F_m u(l_t - l_{t0})}{k_t} \\ y = F_t &= F_m \end{aligned} \tag{2.15}$$

The model can be written in terms of state variables as:

$$\begin{aligned} \dot{x_3} &= \left(\frac{2Ck(Ckx_1 - R_1)\dot{x}_1}{k_t} - \frac{Bk\dot{x}_2 u(kx_2)}{k_t}\right)u(x_3) \\ &= \left(\frac{2Ck(Ckx_1 - R_1)x_2}{k_t} - \frac{Bku(kx_2)}{k_t}\right. \\ &\quad \left.\cdot\left(\frac{mgd}{J}\sin x_1 + \frac{k}{J}((R_1 - Ckx_1)^2 - Bkx_2 - (R_2 + Ckx_1)^2)\right)\right)u(x_3) \end{aligned} \tag{2.16}$$

for the extensor, and

$$
\begin{aligned}
\dot{x_4} &= \left(\frac{2Ck(Ckx_1 + R_1)\dot{x}_1}{k_t} + \frac{Bk\dot{x}_2 u(-kx_2)}{k_t}\right)u(x_4) \\
&= \left(\frac{2Ck(Ckx_1 + R_1)x_2}{k_t} + \frac{Bku(-kx_2)}{k_t}\right. \\
&\quad \left.\cdot\left(\frac{mgd}{J}\sin x_1 + \frac{k}{J}((R_1 - Ckx_1)^2 - Bkx_2 - (R_2 + Ckx_1)^2)\right)\right)u(x_4) \qquad (2.17)
\end{aligned}
$$

for the flexor. The tendon states $x_3$ and $x_4$ are not present in the output. In a strict mathematical sense, the Heaviside step function in (2.15) is not differentiable when its argument is equal to zero. A detailed resolution of these issues will not be elaborated here. For simplicity, we treat the Heaviside function as constant for all time and ignore any issues resulting from the discontinuity. For mathematical justification of this assumption, the reader is referred to Appendix B.

### Golgi Feedback Dynamics

The golgi organs feedback information about the tendon force. Information about the tendon force is very important because it is this force that acts upon the arm itself. The organ's output has a form similar to the spindle sensor's output. The golgi sensors only produce output if the tendon force is above a critical threshold, defined as $\gamma_t$. The sensor output is then raised to a positive fraction, $\epsilon_t$. Note that $\gamma_t$ and $\epsilon_t$ are the corollaries of $\gamma$ and $\epsilon$ from the spindle sensor model. Thus, the golgi signal model is given by:

$$
f_g = (F_t - \gamma_t)(\dot{F}_t)^{\epsilon_t} u(F_t - \gamma_t)u(\dot{F}_t) = (F_m - \gamma_t)(\dot{F_m})^{\epsilon_t} u(F_m - \gamma_t)u(\dot{F_m}) \qquad (2.18)
$$

Each of the golgi signals is then acted upon by a gain constant, $\kappa = 0.00004$, to appropriately weight the force information. The golgi feedbacks are assumed to have the same afferent delay, $\Delta_a$, as the spindle feedback signals. The golgi signals are combined with the spindle signals upon arrival to the spinal cord.

### Updated Spindle Feedback Information

The spindle feedback information given in equation (2.5) is altered by the inclusion of tendons in the model. The musculotendon actuator length and contraction velocity must now be reformulated as:

$$\begin{aligned} l_1 &= l_{ex} + l_{tex} = kx_1 + x_3 \\ l_2 &= l_{fl} + l_{tfl} = kx_1 + x_4 \end{aligned} \tag{2.19}$$

and

$$\begin{aligned} \dot{l_1} &= -k\dot{x}_1 + \dot{x}_3 = -kx_2 + \frac{(R_1 - Ckx_1)^2 - Bkx_2u(kx_2)}{k_t} \\ \dot{l_2} &= k\dot{x}_1 + \dot{x}_4 = kx_2 + \frac{(R_2 + Ckx_1)^2 + Bkx_2u(-kx_2)}{k_t} \end{aligned} \tag{2.20}$$

### Tendon Filtering Effects

It has been shown the tendons attenuate signals with significant high frequency components and only pass signals with frequency content below a certain cutoff. The cutoff frequency is related to the stiffness of the tendon itself, and is consequently a function of the spring constant, $k_t$ [15]. The filter acts on the state variables $x_3$ and $x_4$, and subsequently on the musculotendon actuator lengths and contraction velocities $l_1$, $l_2$, $\dot{l}_1$, and $\dot{l}_2$, respectively. In the Laplace domain with complex frequency variable, $s = \sigma + i\omega$, the filter is given by the transfer function,

$$G(s) = \frac{k_t}{2400s + k_t} = \frac{20}{s + 20} \tag{2.21}$$

or

$$\begin{bmatrix} \dot{x_5} \\ \dot{x_6} \end{bmatrix} = \begin{bmatrix} -20 & 0 \\ 0 & -20 \end{bmatrix} \begin{bmatrix} x_5 \\ x_6 \end{bmatrix} \tag{2.22}$$

where $x_5$ and $x_6$ are the filter state variables of the extensor and flexor actuators, respectively.

### 2.2.1 The Complete Sixth Order System

The complete sixth order system is defined by the state equations,

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{mgd}{J}\sin x_1 + \frac{k}{J}((R_1 - Ckx_1)^2 - Bkx_2u(kx_2) - (R_2 + Ckx_1)^2 - Bkx_2u(-kx_2)) \\
\dot{x}_3 &= \left(\frac{2Ck(Ckx_1 - R_1)x_2}{k_t} - \frac{Bku(kx_2)}{k_t}\right. \\
&\quad \left.\cdot\left(\frac{mgd}{J}\sin x_1 + \frac{k}{J}\Big((R_1 - Ckx_1)^2 - Bkx_2 - (R_2 + Ckx_1)^2\Big)\right)\right)u(x_3) \\
\dot{x}_4 &= \left(\frac{2Ck(Ckx_1 + R_1)x_2}{k_t} + \frac{Bku(-kx_2)}{k_t}\right. \\
&\quad \left.\cdot\left(\frac{mgd}{J}\sin x_1 + \frac{k}{J}\Big((R_1 - Ckx_1)^2 - Bkx_2 - (R_2 + Ckx_1)^2\Big)\right)\right)u(x_4) \\
\dot{x}_5 &= -20x_5 \\
\dot{x}_6 &= -20x_6
\end{aligned} \tag{2.23}$$

and the output matrix equation,

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \tag{2.24}$$

where $R_1(x,u)$ and $R_2(x,u)$ are given by,

$$\begin{aligned}
R_1(x,u) &= u_1(f_s + f_g) + u_1 \\
R_2(x,u) &= u_2(f_s + f_g) + u_2
\end{aligned} \tag{2.25}$$

The spindle feedback $f_s(x, u)$ and the golgi feedback $f_g(x, u)$ are complicated functions of the state and control. A system diagram is presented below.

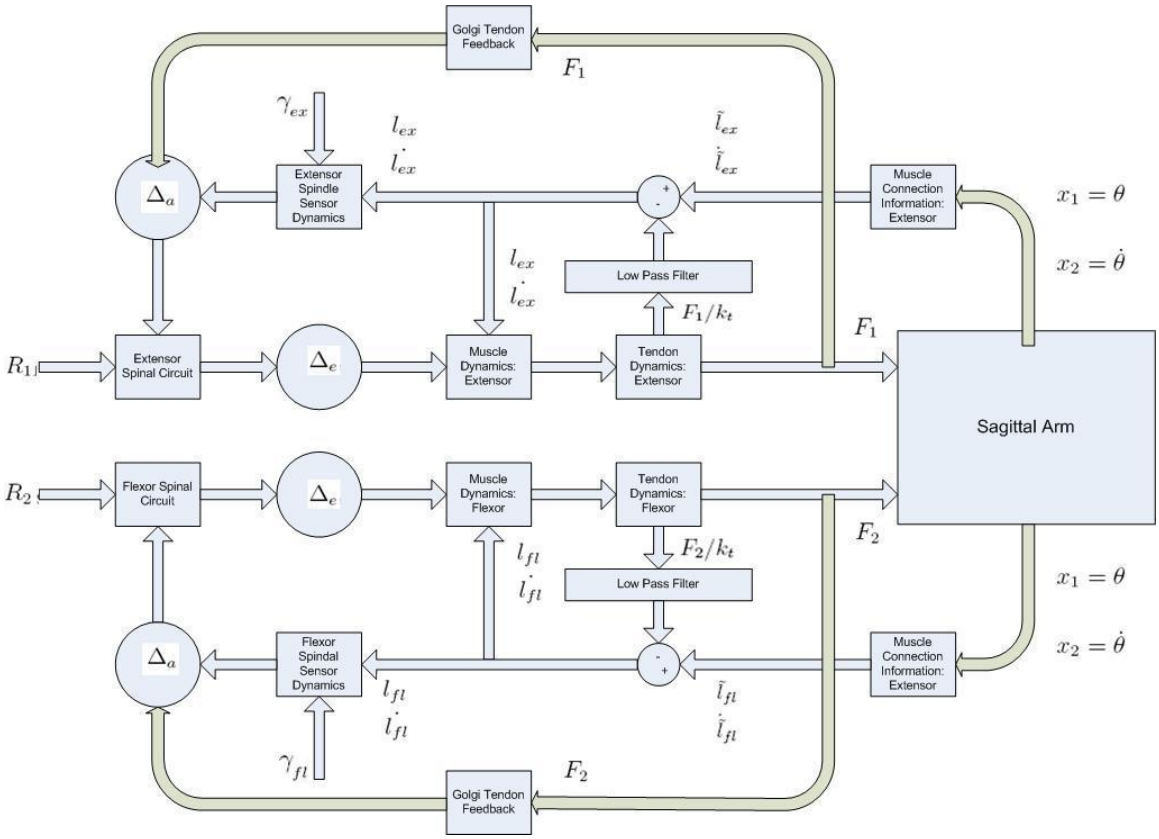


Figure 2.2: The sixth order model has the inputs $\gamma_{ex}$, $\gamma_{fl}$, $R_1$, and $R_2$ which correspond the extensor and flexor threshold lengths and the neural firing inputs, respectively. The model has the angular position $x_1 = \theta$ and angular velocity $x_2 = \dot{\theta}$ as the system output. The afferent and efferent delays are denoted as $\Delta_a$ and $\Delta_e$, respectively.

A table of nominal values of system parameters that are used in simulation is given below:

| Parameter | Definition | Value |
|---|---|---|
| B | muscle Hill effect viscous constant | 75 |
| C | muscle passive elasticity constant | 30 |
| d | distance from arm's center of gravity to hinge | 0.18 [m] |
| g | gravity constant | 10 [m/$s^2$] |
| J | moment of inertia of arm about hinge | 0.075 [kg·$m^2$] |
| k | actuator moment arm | 0.04 [m] |
| $k_t$ | tendon spring constant | 48000 [N/m] |
| $l_0$ | unstretched muscle fiber length | 0.3 [m] |
| m | mass of arm | 1.9 [kg] |
| q | spindle sensor gain | 2 |
| $\Delta_a$ | afferent delay | 0.005 [s] |
| $\Delta_e$ | efferent delay | 0.035 [s] |
| $\epsilon$ | spindle sensor contraction velocity exponent | 0.2 |
| $\epsilon_t$ | golgi sensor force derivative exponent | 0.033 |
| $\gamma$ | muscle threshold length | 0 [m] |
| $\gamma_t$ | tendon threshold force | 0 [N] |
| $\kappa$ | golgi feedback signal gain | 0.00004 |

Table 2.2: Parameters of the Sixth Order Model

## 2.3 Summary

In this chapter two models of the musculoskeletal arm were developed: a two dimensional model and a six dimensional model. Each of these models is based on a one-link arm that is hinged to an inertial reference frame. In the second order model, the arm is equipped with nonlinear muscle-like actuators and spindle-like feedback sensors. The model incorporates the unidirectional nature of muscle forces by restricting such forces to positive numbers. Thus, just as in natural muscle tissue, the muscle model can only pull in the direction of shortening. The neural feedback signals associated with spindle-like sensors are also restricted to be positive. The positiveness of the neural signals allows them to act as neural firing rates.

The sixth order model is an enhancement of the second order model. In addition to the previously described features, this model includes tendon dynamics based on a Hookeian spring, and golgi feedback organs. The golgi organs feedback information about the force acting on the arm. These feedback signals are also only allowed to be positive, so as to preserve the signal's relationship with natural neural firing rates.

# Chapter 3

# Stability Properties and Point-to-Point Movement of the Sagittal Arm

Before we delve into more complicated issues, it is important to verify that the models we have created satisfy some basic requirements. If the arm model does not share basic properties of its biological counterpart, then it cannot be employed for further investigation. In study of the control of arm movements one of the most important features of the arm is stability about an equilibria. The least we should expect from the model is the ability to maintain an arbitrary angular position (an equilibrium). Further, in the presence of small perturbations, the arm should resettle to the desired position after a short time.

In addition to stability analysis, simple point-to-point movements of the sagittal arm will be analyzed for both the second and sixth order models. The strategy in point-to-point movement is the same as in Dinneen and Hemami. The movement has two phases. First, the system is brought out of the equilibria at the origin ($x_1 = x_2 = 0$) by raising and lowering the coactivation levels to appropriate values. After this phase is complete, the system inputs are set to appropriate constant values for the duration of the movement.

Although a formal stability analysis (i.e. a Lyapunov analysis) of the sixth order model is

not performed due to the complexity of the system, system stability is confirmed implicitly for the selected cases through successful demonstration of point-to-point movement.

## 3.1 Stability of the Second Order Model

First we verify the second order model. If we neglect the spinal feedback signals the system model of equation (C) becomes:

$$f_1(x) = x_2$$

$$f_2(x) = \frac{mgd}{J}\sin x_1 + \frac{k}{J}\Big((u_1 - Ckx_1)^2 - Bkx_2 - (u_2 + Ckx_1)^2\Big)$$

If we set $u_1 = u_2 = u$ the system reduces to,

$$f_1(x) = x_2$$

$$f_2(x) = \frac{mgd}{J}\sin x_1 - \frac{k}{J}(4uCkx_1 + Bkx_2) \tag{3.1}$$

If we restrict our region of interest to the neighborhood around the equilibrium $x = (0, 0)$ the system is well approximated by

$$\ddot{\theta} + \frac{k^2B}{J}\dot{\theta} + \Big(\frac{4Ck^2u}{J} - \frac{mgd}{J}\Big)\theta = 0$$

and a necessary and sufficient condition for the equilibrium's stability is noted as [2]:

$$4Ck^2u > mgd$$

We test the model by considering the case where the actuators are coactivated with $u = 25$ and the system has initial conditions $x_1 = 0$ and $x_2 = 0.25$. Thus, the system is at rest at the origin and is perturbed by a disturbance. The results of the simulation, plotted below, show that the system does indeed return to the origin.

A lucid illustration of the trajectory converging to the origin can be seen in the system phase portrait. The speed in which the system stabilizes is a function of the Hill effect viscous constant, $B$, which defines the damping in the system. It can be shown the system

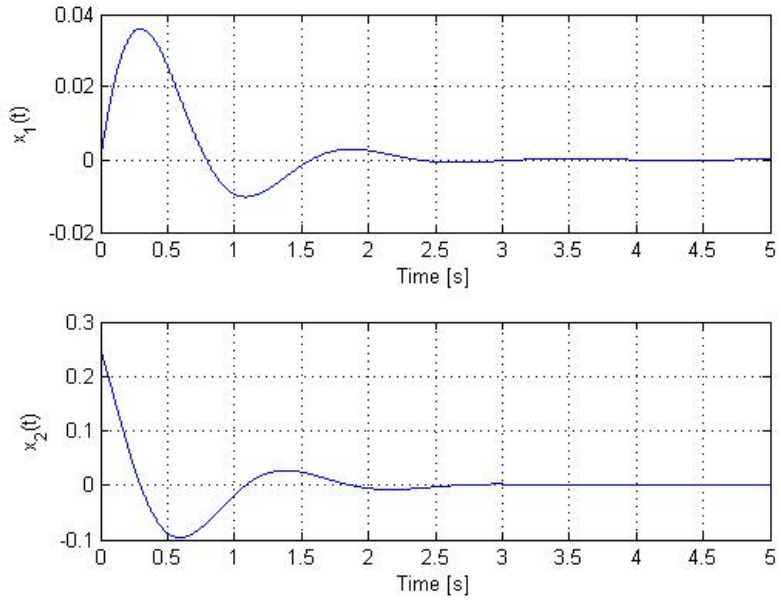


Figure 3.1: Second Order Model Equilibrium Stability, State Trajectories

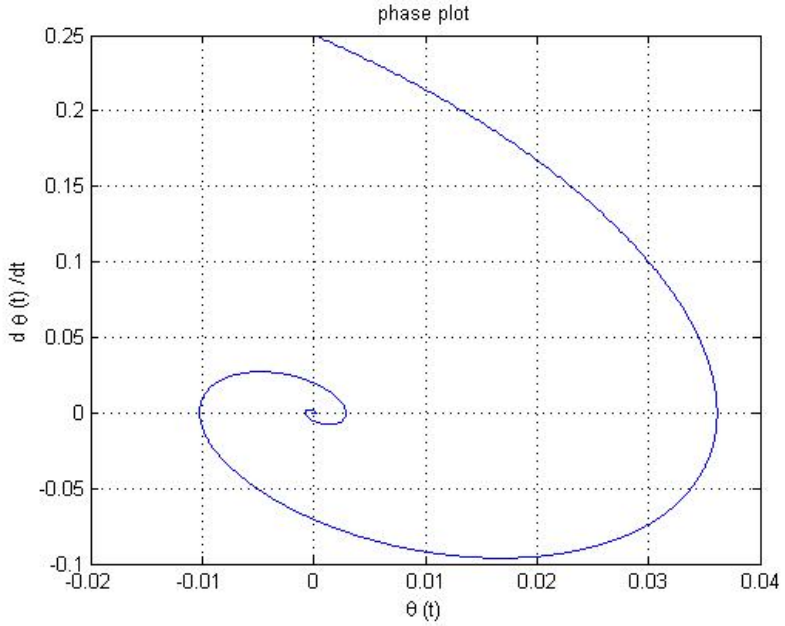


Figure 3.2: Second Order Model Equilibrium Stability, Phase Portrait

will stabilize at an arbitrary angular position $x_1 \in \begin{bmatrix} 0 & \pi \end{bmatrix}$ provided suitable $u_1$ and $u_2$ are selected.

If the spindle feedback is turned on, but still neglecting afferent and efferent signal delays, the system's rate of convergence to a stable equilibrium increases. Repeating the same experiment, the system settles to the origin more quickly than in the previous case. The phase portrait is given below.

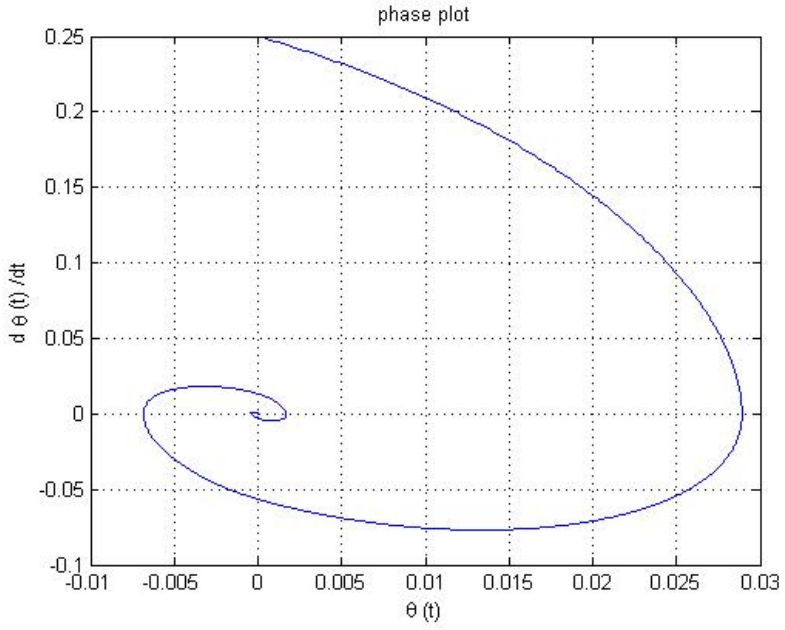


Figure 3.3: Second Order Model Equilibrium Stability with Spindle Feedback, State Trajectories

## 3.2 Stability of the Sixth Order Model

Next, we proceed with analysis of the sixth order model. In the absence of golgi and spindle feedback signals, the addition of tendons to the model does not effect the system's ability to stabilize at an fixed angular position. This result is readily apparent from the input output equation of the tendon model (2.15), in which the tendon force is equal to the muscle force. Thus, the addition of linear tendons has effectively left the system unaltered, and has not qualitatively changed any of its properties.

However, the system is altered if either spindle or golgi feedbacks are present. Additional simulations are performed to test the model with the adjusted spindle, golgi, and both adjusted spindle and golgi feedback. If our model is constructed properly is should still possess the desired stability property, namely, convergence to a fixed angular position given coactivation of the actuators and constant neural firing rates. We will test these properties in the next section by performing point-to-point movement. Our goal is to implicitly demonstrate the stability of equilibria in the sixth order system.

## 3.3 Stable Point-to-Point Movements

We follow the point-to-point movement strategy of Dinneen and Hemami. In this strategy the movement consists of two separate phases: an initiation phase and a steady state phase. In the initiation phase the system is brought from zero initial position and velocity to suitable intermediate velocity which is known *a priori* from trial and error or learning [2]. For a given desired angular position, the duration of the initiation phase is arbitrary and depends on how 'fast' of a movement is desired. A shorter duration requires a higher tension and therefore a higher activation level. When the initiation phase is concluded, the steady state phase begins and the inputs to the model are set to appropriate constants. This second phase lasts for the duration of the simulation.

We perform point-to-point simulations for the second order model with and without spindle feedback. Additionally, we simulate three cases of the sixth order model: with spindle feedback, with golgi feedback, and with both spindle and golgi feedback. In all of our simulations we fix the values of the spindle thresholds, $\gamma_1$ and $\gamma_2$, to zero. Similarly, in simulations with the six dimensional model we set the thresholds of the golgi organ feedback, $\gamma_{t1}$ and $\gamma_{t2}$, to zero for all time. We also continue to neglect both afferent and efferent delays.

In order to distinguish the effects of the various types of feedback signals on the system response we perform all point-to-point motion simulations with the same neural firing rates. In keeping with our motion strategy, the neural rate signals are defined as,

$$
\begin{aligned}
R_1 &= 2.7\sin(12.5\pi t)u(-t+0.04) + 25u(-t+0.04) + 8u(t-0.04) \\
R_2 &= -1.7\sin(12.5\pi t)u(-t+0.04) + 25u(-t+0.04) + 6u(t-0.04) \qquad (3.2)
\end{aligned}
$$

where $u(t)$ is the Heaviside step function. Our definition causes the initiation phase activation levels of the extensor and flexor to be raised and lowered to 27.7 and 23.3, respectively. The initiation phase is assumed to last for 40 ms. The extensor and flexor steady state actuation values are selected as 8 and 6, respectively. The neural firing rate input signals, $R_1$ and $R_2$, are plotted in Figure (3.4).

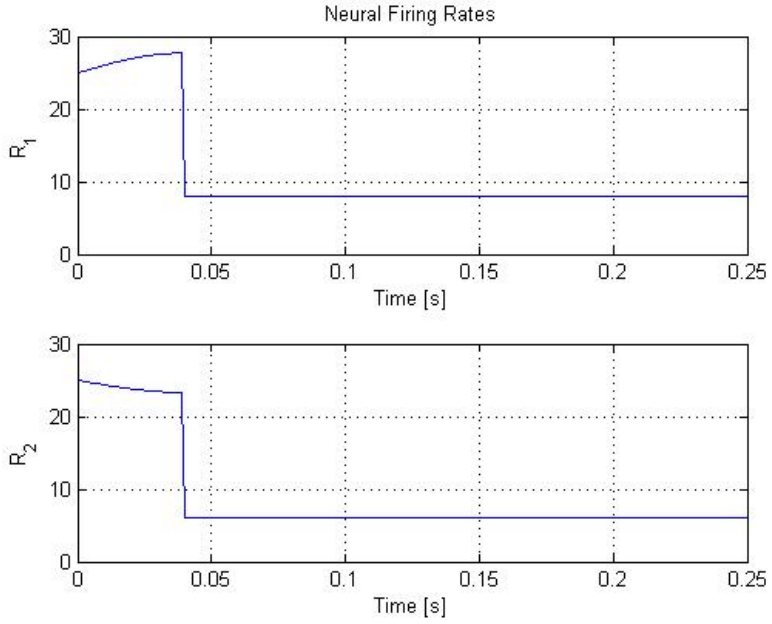


Figure 3.4: Point-to-Point Motion: Neural Firing Inputs $R_1$ and $R_2$. Note that the inputs hold their respective steady state values of 8 and 6 until the simulation terminates at $t = 5$ s. Here we truncate the plot window to clearly depict the input's initiation phase behavior.

### 3.3.1 Point-to-Point Movements of the Second Order System

The system without spindle feedback has the trajectory and phase plot given in Figure (3.5) and Figure (3.6), respectively. The trajectory and phase plot for the system with spindle

feedback are depicted in Figure (3.7) and Figure (3.8). Recall that each of these simulations is performed with $\gamma_1 = \gamma_2 = 0$, and with $R_1$ and $R_2$ as in equation (3.2).

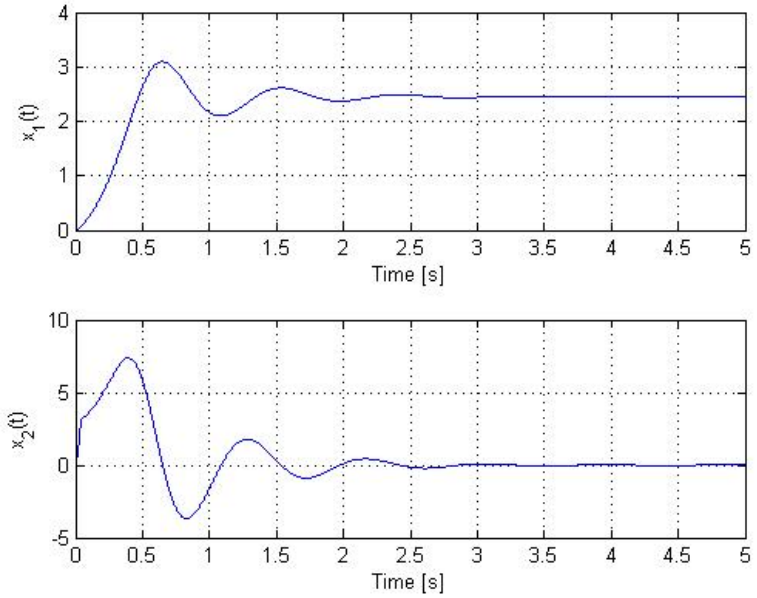


Figure 3.5: Point-to-Point Movement: Second Order Model without Spindle Feedback, State Trajectories

Examination of the four figures allows us to make some qualitative observations. First, we notice that the inclusion of spindle feedback significantly decreases the settling time to the state equilibrium. The spindle feedback also has a damping effect and reduces the overshoot from about 25% in Figure (3.5) to about 16% in Figure (3.8). The inclusion of spindle feedback appears to have little effect on the system rise time or steady state behavior. From the above plots, it is clear that point-to-point motion of the second order model has its best response with spindle feedback activated.

### 3.3.2 Point-to-Point Movements of the Sixth Order System

The system with spindle and golgi feedback has the trajectory and phase plot given in Figure (3.9) and Figure (3.10). The extensor and flexor actuator forces are also plotted for this simulation in Figure (3.11). The trajectory and phase plot for the system with only

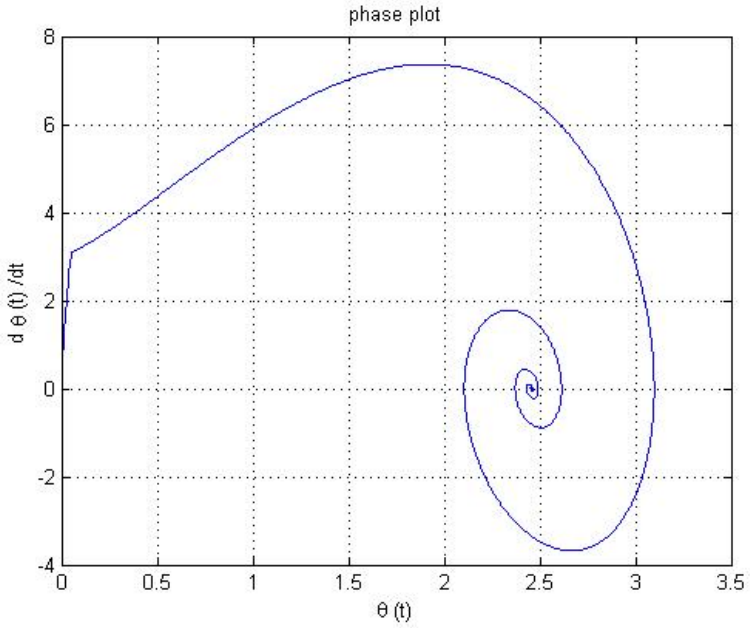


Figure 3.6: Point-to-Point Movement: Second Order Model without Spindle Feedback, Phase Portrait

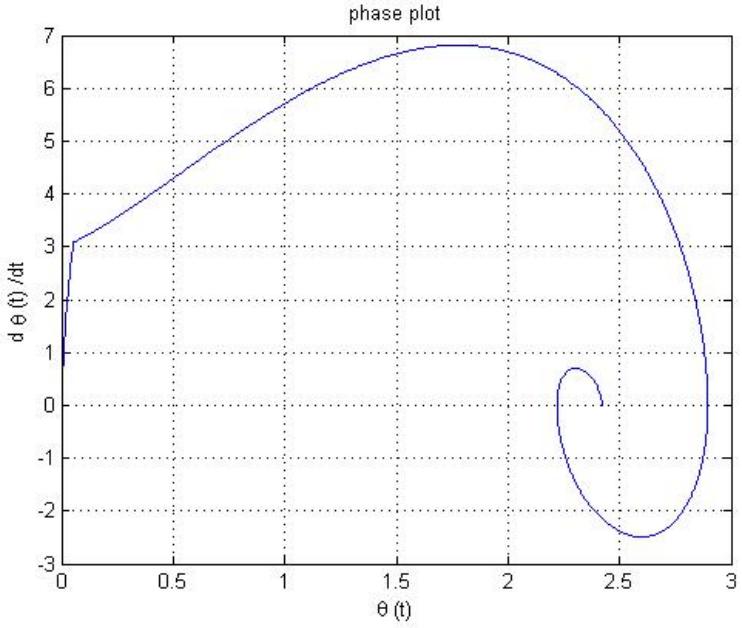


Figure 3.7: Point-to-Point Movement: Second Order Model with Spindle Feedback, Phase Portrait

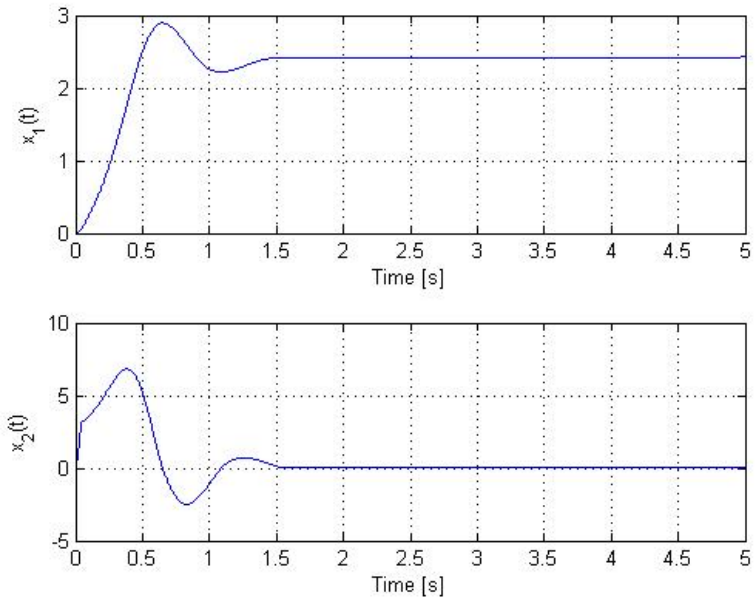


Figure 3.8: Point-to-Point Movement: Second Order Model with Spindle Feedback, State Trajectories

spindle feedback are depicted in Figure (3.12) and Figure (3.13), and with only golgi feedback in Figure (3.14) and Figure (3.15), respectively. Recall that each of these simulations is performed with $\gamma_1 = \gamma_2 = \gamma_{t1} = \gamma_{t2} = 0$, and with $R_1$ and $R_2$ as in equation (3.2).

In all cases the arm stabilizes at a fixed angular position. The rate of convergence varies between each of the three cases. From our analysis it is apparent the golgi feedback plays a role in speeding up the system dynamics. The system with only spindle feedback has a larger rise time than either of the other two systems, and the systems with golgi feedback are approximately 25% faster, in terms of rise time, than the system with only spindle feedback.

Similar to the second order case, the spindle feedback has a damping effect on the system dynamics. The system with only golgi feedback is much more oscillatory than either of the other two systems. The golgi-only system has the largest settling time, which is significantly longer than the other systems. The systems with spindle feedback have percent overshoots that are around 20% less than the golgi-only system.

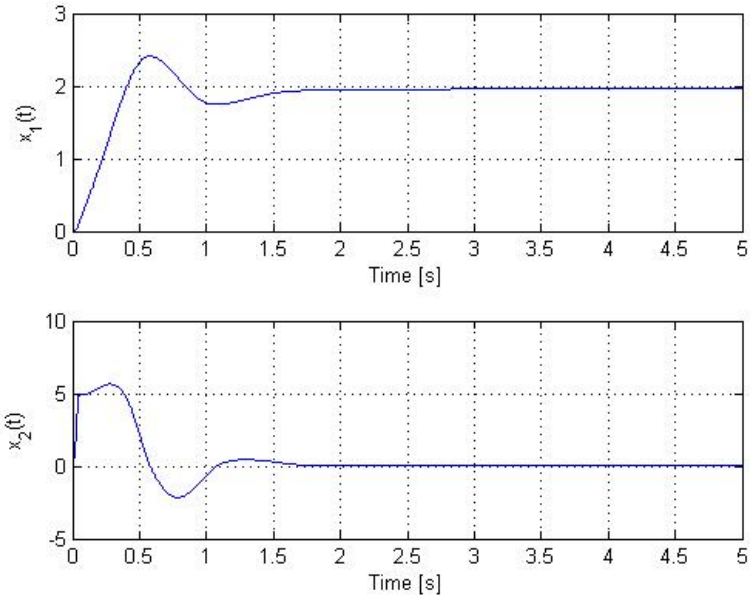


Figure 3.9: Point-to-Point Movement: Sixth Order Model with Spindle and Golgi Feedback, State Trajectories

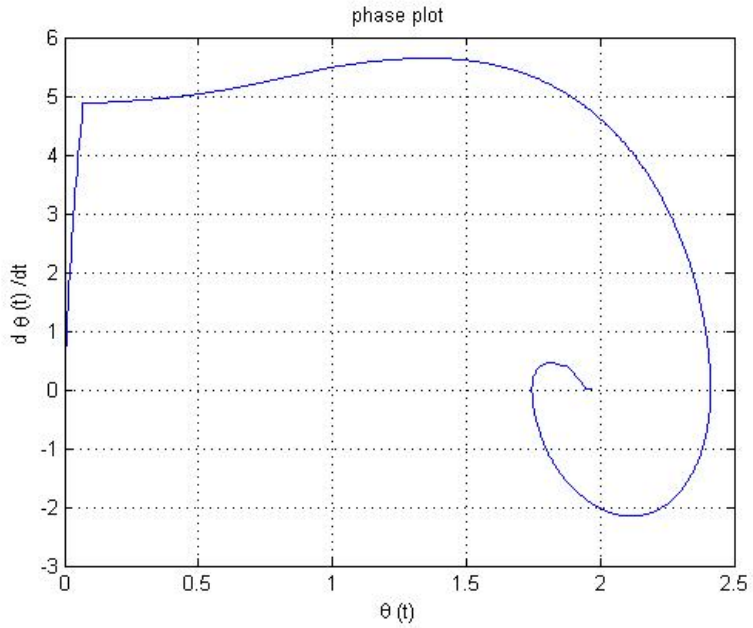


Figure 3.10: Point-to-Point Movement: Sixth Order Model with Spindle and Golgi Feedback, Phase Portrait

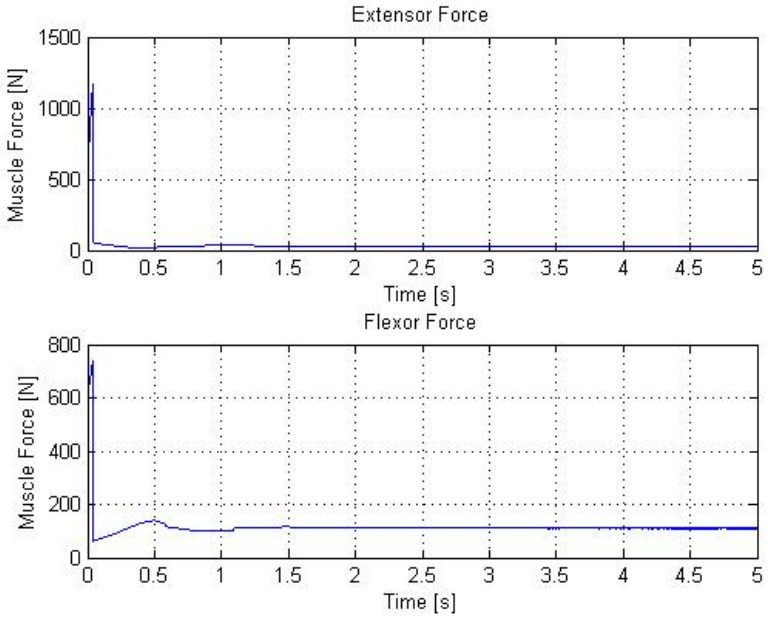


Figure 3.11: Point-to-Point Movement: Sixth Order Model with Spindle and Golgi Feedback, Muscle Forces

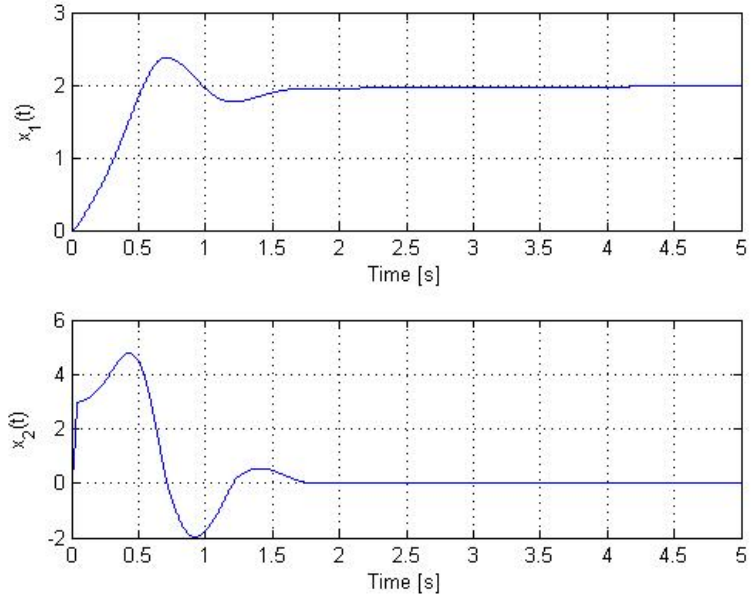


Figure 3.12: Point-to-Point Movement: Sixth Order Model with Spindle Feedback, State Trajectories

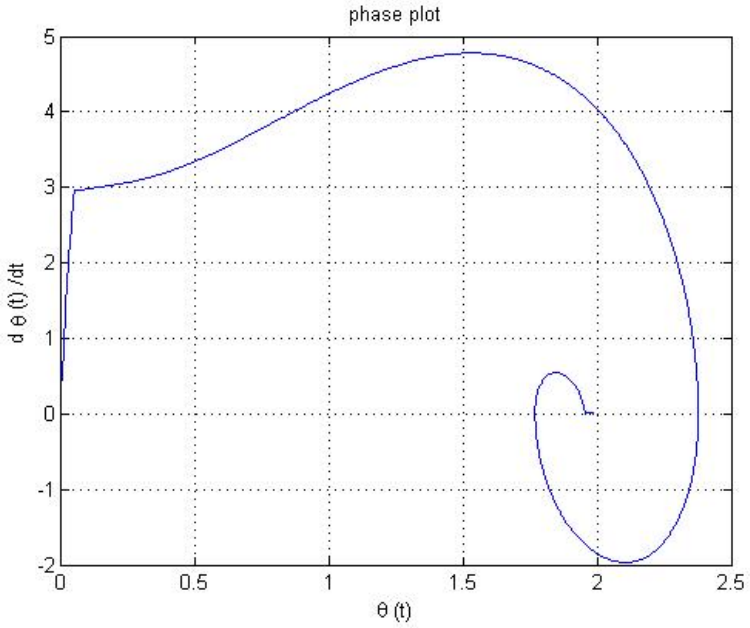


Figure 3.13: Point-to-Point Movement: Sixth Order Model with Spindle Feedback, Phase Portrait

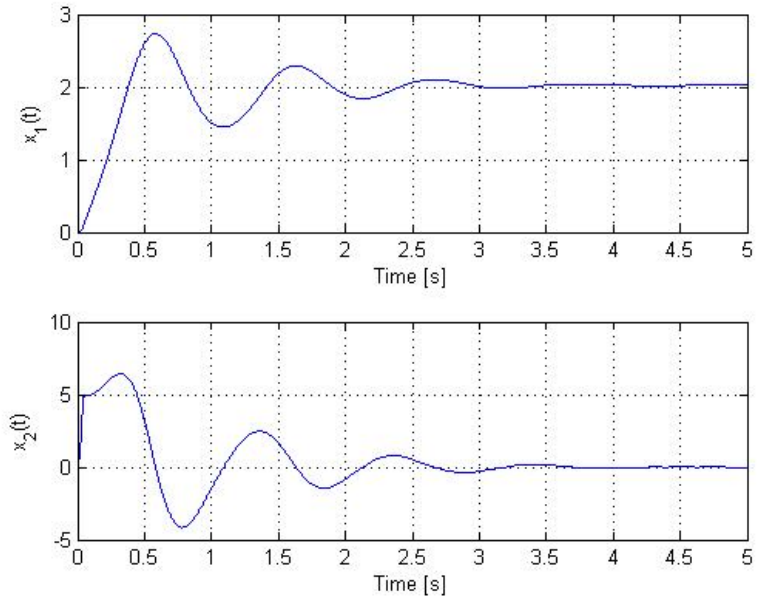


Figure 3.14: Point-to-Point Movement: Sixth Order Model with Golgi Feedback, State Trajectories

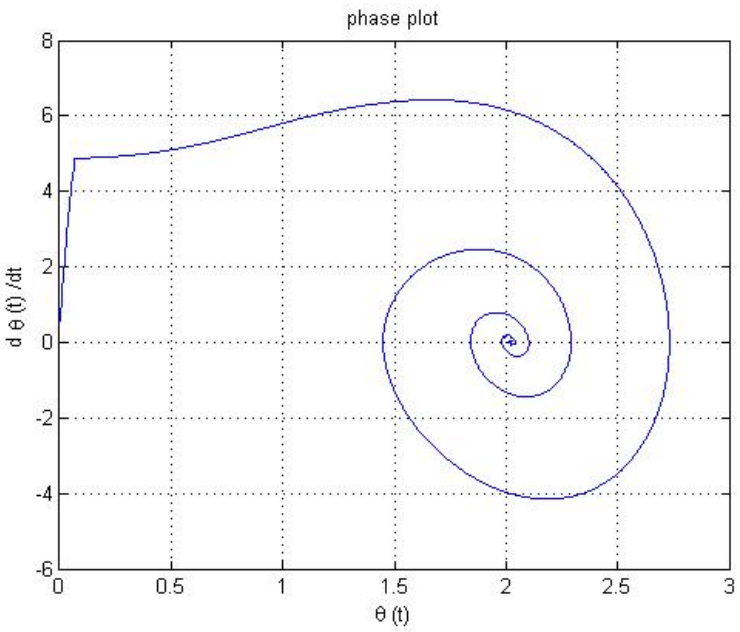


Figure 3.15: Point-to-Point Movement: Sixth Order Model with Golgi Feedback, Phase Portrait

## 3.4 Summary

Upon completion of our stability analysis it is apparent that the model embodies the dynamics of the arm well enough to be used in a study of neurological control. It was shown that a respective equilibrium point of the second order model is stable if the system is coactivated at a sufficiently high level. In a like manner, stability of the sixth order model was implicitly shown in select cases by successful point-to-point movement. The successful point-to-point movements carried out by both models speak to their utility in further experimentation.

The effects of spindle and golgi feedbacks on the system response were also investigated. The spindle feedback was found to have a large influence on the system's damping characteristics. Inclusion of spindle feedback corresponded to improved percent overshoots and settling times. The inclusion of golgi feedback was found to decrease the system rise time, thereby making the system faster. Interestingly, each type of feedback modulates a different type of performance metric, allowing the central nervous system great flexibility in governing locomotion.

Overall, the effect of the tendon itself is equivalent to low pass filtering with intrinsic force and force derivative feedback [6]. The tendon also reduced the final equilibrium position from 2.5 rad, for the second order model's simulations, to 2 rad for simulations with the sixth order model. It should also be noted that in all cases coactivation of the actuators is an essential stability requirement.

# Chapter 4

# Formulation of a Control Problem for Complex Motion

Now that the requisite stability properties of the models have been confirmed, we shift our focus to the control of the arm in more complicated sagittal motion. Although there are a vast number of possible motions to simulate, we will consider the case of periodic continuous motion as a crude representation of dancing. The periodic motion of the arm approximates a natural reaction to the beat of a piece of music. Just as dancing follows the rhythm of music, our control objective will be for the arm to track a periodic signal.

Several theories exist as to how the brain orchestrates such motion. The simplest of these theories employ sensory (i.e. visual) or spinal feedback signals in the composition of the control law. However, such models are inapplicable for control of rapid movement due to substantial delays in afferent signal paths. These delays, typically lasting around 40 ms, necessitate a more complex model. A comparatively recent development has been the formulation of the internal model theory of motor control [4]. Under this hypothesis, the brain produces an inverse model or models based on the planned trajectory and uses them to calculate the appropriate control signals in a purely feedforward fashion [4].

We seek to employ a hybrid version of these theories. While we employ feedback in the derivation of our control law, the feedback is impeded by delays of length similar to that

of the physiological system itself. Thus, our control philosophy does not conflict with the aforementioned biological constraints. In keeping with the internal model theory, we seek a controller that takes a desired trajectory as input. Such a structure can be achieved via an optimum control design.

The formulation of locomotive trajectory planning as an optimal control problem is a well established technique in biomechanical systems [4],[14]. Formulating the control problem in such a manner allows for the minimization of desired cost functions while providing accurate tracking of a reference input. Interestingly, the necessity of *a priori* knowledge of a desired state trajectory is not an issue for the biological control model. Although obtaining such knowledge is often problematic – if not impossible in other systems, the requisite information is always available for trajectory planning. Intuitively speaking, it is obvious that one 'knows' where one would like to move before actually doing so. The irrelevance of numerical efficiency is equally fortuitous. Because we are only interested in the qualitative behavior of solutions, we can calculate the state feedback gain matrices online, without having to worry about computation time.

Despite its advantages, the optimal control formulation has some shortcomings. Because the system is highly nonlinear, the control law derived via Jacobian linearizations will always be suboptimal. If a computer is used to numerically solve the discrete time Riccati equation, the best available approximation is derived by repeatedly linearizing the system at each time step. The accuracy of the linearization is then a function of the sampling interval and the dynamics of the system at the respective operating point. The sampling time of 1 ms used in the model is small enough to ensure a reasonable degree of accuracy.

Another significant issue is the strategy's lack of robustness to delays. As was previously noted, the model contains significant delays associated with the spinal feedback signals. If the delays are ignored, the control law derived at a particular time interval losses accuracy, and may not achieve its objective. To rectify the situation in future research, delays may be modeled with Padé approximations. Fortunately, it turns out that the strategy performs well enough for delays in the order of interest.

An attractive alternative is the tracking control problem of a feedback linearized system. In this formulation, a linearizing feedback reduces the complexity of control task to the manipulation of a linear system. Thus, a controller designed with a feedback linearized model would, in theory, outperform the suboptimal control derived with Jacobian linearization. As appealing as this approach seems, the technique is demonstrated to be invalid for the models previously presented.

In all cases, we restrict the reference trajectory signal $z(t)$ to the class of $C^3$ functions. That is, we require the first and second signal derivatives to be strictly continuous in $t$.

## 4.1 A Discrete Time Sub-Optimal Control Problem

The optimal strategy is designed to minimize the arm's end point variance relative to the desired angular position and velocity at each time step. While variance itself is a purely kinematic variable, the process of muscular actuation is dynamic. Thus a cost function defined by end point variance takes into account both energy and position based constraints simultaneously [4]. The cost function may be defined in continuous time as:

$$J = \int_{t_0}^{T} w_t(x) \| x(t) - x_{ref}(t) \| dt \tag{4.1}$$

Here $t_0$ is the start time of the movement and $T$ is the movement's end time. The vector function $x(t)$ is the states $\theta = x_1$ and $\dot{\theta} = x_2$ as defined in chapter 2, namely the angular position and velocity of the arm's endpoint. The weight function $w_t(x)$ may be chosen to emphasize the relative importance of minimizing $J$ with respect to one variable over another. For our purposes, we assume the weight function is a constant. Our choice follows from the interrelation of angular position and velocity.

In order to derive a computer-based solution to the optimization problem (e.g. a recursive process) the cost function and system must be discretized. In the case of the cost function we ignore discretization issues and simply replace the continuous argument $t$ with the discrete argument $t_k$. We are free to do so because the cost function is our own mathematical

construction. However, we will have to be considerable more careful in our discretization of the state space model (presented in the next section). The cost function can be represented in discrete time as:

$$J = \sum_{j=T_k+1}^{T_k+N+1} w_t[x] \| x[t_k] - x_{ref}[t_k] \| \tag{4.2}$$

Here the $T_k$ is the first time step of motion and $N$ is the duration of motion in time steps [11].

Keeping the control goal in mind (tracking a trajectory based on a periodic input that is known *a priori*), the input sequence should approximate the beat of piece of music. We might think of the music's beats as the respective extrema of the periodic signal. Our periodic reference input signal will be denoted as $z(t)$.

Before we develop the control problem in more detail we pause to linearize and discretize the second order model. These operations will prove essential in the application of our optimal control strategy.

## 4.2 Linearization of the Second Order Model

In order to achieve our control objective we must transform the system of equation (C) into a more workable form. Accurate reference tracking of our system is quite a difficult problem because no satisfactory analytical theory exists for the general nonlinear system. Thus, results for the derivation of control laws for such systems are extremely limited. While we were able to work with the nonlinear system directly in our analysis of simple point-to-point motion, forcing such a system to follow more complicated motions is a daunting task. As such, our first objective will be to derive an appropriate linearization of the second or system. With a linearized model we can apply the powerful, well defined methods of linear feedback control.

The dynamics of the nonlinear second order system are given by (C):

$$\dot{x_1} = x_2$$

$$\dot{x_2} = \frac{mgd}{J}\sin x_1 + \frac{k}{J}\Big((R_1 - Ckx_1)^2 - Bkx_2u(kx_2) - (R_2 + Ckx_1)^2 - Bkx_2u(-kx_2)\Big)$$

The state equations can be rewritten in a simpler form making use of the fact that

$$-Bkx_2(t) = -Bkx_2(t)u(kx_2(t)) - Bkx_2(t)u(-kx_2(t))$$

which leaves us with

$$f_1(x) = \dot{x_1} = x_2$$

$$f_2(x) = \dot{x_2} = \frac{mgd}{J}\sin x_1 + \frac{k}{J}((R_1 - Ckx_1)^2 - Bkx_2 - (R_2 + Ckx_1)^2) \tag{4.3}$$

The above simplification follows from our muscle model, as the involved terms correspond to the Hill effect. Recall that Hill effect describes a muscle's tendency to exert less tension under greater contraction velocities. The simplification results from the fact that only one muscle can have positive *contraction* velocity at any given time under sagittal motion. Moreover, at least one muscle must be contracting (resulting in a nonzero Hill effect) at any given time that the arm is moving (e.g when $x_2 \neq 0$). Thus, the above expression is identically, $-Bkx_2(t)$.

We know from the Hartman-Grobman Theorem that in the neighborhood of a hyperbolic equilibrium the system can be modeled by a set of linear differential equations [5] [7]. Thus, we assume that close to the hyperbolic equilibrium at the origin our model can be well approximated by linear equations. If we take the Jacobian of the system with respect to $x$ and $u$, we arrive at the following state space model.

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\Big|_{x=x_{op},u=u^*} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}\Big|_{x=x_{op},u=u^*} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$
$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{4.4}$$

A detailed derivation of these matrices, with a complete functional state space description is relegated to Appendix C.

## 4.3 The Discretized System

Now that we have constructed a linear model, we able to apply the powerful methods of linear feedback control to achieve our objectives. However, in order to apply the desired optimal control strategy, one has to solve the Riccati matrix equation to obtain the optimal feedback gain matrix. A quick inspection of the Riccati equation, given below, shows that deriving a symbolic solution for $K$ is a real mathematical challenge.

$$A^T S + SA - SBR^{-1}B^T S + Q = 0$$

The feedback gain matrix $K$ is then derived as:

$$K = R^{-1}B^T S$$

Note that this equation must be solved for all values of the system's $A$ and $B$ matrices, derived through Jacobian linearization. When one considers that fact that these matrices are inherently time variant, because $x(t)$ and $u^*(t)$ are time variant, hopes of obtaining a solution by hand quickly dissipate. Thus, it is essential to discretize the system in order to have a computer solve the Riccati equation for us.

We discretize the system by sampling it at a rate of $1/T = 1000$ Hz. Setting $dx/dt = [x(t_{k+1}) - x(t_k)]/T$, and substituting $t_k$ for $t$ in the right hand side of (4.4) we arrive at:

$$\begin{aligned} x[t_{k+1}] &= [I + TA|_{x=x_{op},u=u^*}]x[t_k] + TB|_{x=x_{op},u=u^*}u[t_k] \\ y[t_k] &= Cx[t_k] \end{aligned} \tag{4.5}$$

or

$$\begin{bmatrix} x_1[t_{k+1}] \\ x_2[t_{k+1}] \end{bmatrix} = \begin{bmatrix} 1 & T \\ \frac{mgd}{J}\cos x_1 + \frac{2k^2C}{J}(R_1 + R_2 + 2Cl_0) & \frac{Tk^2B}{J} \end{bmatrix}\Bigg|_{x=x_{op},u=u^*} \begin{bmatrix} x_1[t_k] \\ x_2[t_k] \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 \\ \frac{-2Tk}{J}(R_1 + Cl_0 - Ckx_1) & \frac{2Tk}{J}(R_2 + Cl_0 - Ckx_1) \end{bmatrix} \Bigg|_{x=x_{op},u=u^*} \begin{bmatrix} u_1[t_k] \\ u_2[t_k] \end{bmatrix}$$

$$\begin{bmatrix} y_1[t_k] \\ y_2[t_k] \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1[t_k] \\ x_2[t_k] \end{bmatrix} \tag{4.6}$$

The simplicity of the above discretization process is allowable only because it is assumed that the state $x[t_k]$ and the control input $u[t_k]$ are well defined functions for all $t_k$. Equation (4.6) is the system model that will be used to generate the control law.

## 4.4 Sub-Optimal Control of the Discretized Linearized Second Order Model

The desired outcome is for the arm to successfully track a pre-specified trajectory in the state space, namely $z(t)$. This is equivalent to the arm being forced to equilibrium at $\Delta x = x(t) - z(t) = 0$ [5]. The output feedback employed by the controller will be the angular position and angular velocity errors. The control inputs to the model will be the neural outputs $R_1$ and $R_2$, respectively. Although, $R_1$ and $R_2$ are not the neural firing rates, the two inputs are related to the firing rates by equation (2.6). The control input is then $u(x,t) = [\ R_1(x,t) \quad R_2(x,t)\ ]^T]$.

In order to maintain a high degree of accuracy, the operating points for the linearization of the system's $A$ and $B$ matrices will be updated at every time step $t_k$. Although this method is computationally intensive, it allows for the dynamics of the nonlinear system to be encapsulated by the linearized model as well as possible. Additionally, the 1 ms sampling time is slow enough to ensure the system dynamics do not change dramatically over a given sampling interval.

Placing the system in $\Delta x$ coordinates allows for the end point variance to be minimized simply by minimizing the state. Thus, the problem can be solved by a Linear Quadratic Regulator [9][10]. Revisiting the cost function of equation (4.2), we are now ready to set up

the sub-optimal control in the sense of a quadratic performance index. Defining $\Delta x[t_k] = x[t_k] - x_{ref}[t_k] = x[t_k] - z[t_k]$ the cost function can be written as

$$J = \sum_{j=T_k+1}^{T_k+N+1} \Delta x^T Q \Delta x + u^T R u \tag{4.7}$$

Where $Q$ and $R$ are positive definite matrices given by:

$$Q = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Here we emphasize the importance of minimizing the state tracking error over minimizing the control gains. In the new $\Delta x$ coordinates, the state equation is:

$$x[t_k + 1] = PAP^{-1}x[t_k] + PBu[t_k] \tag{4.8}$$

$$\Delta x[t_k + 1] = A_\Delta \Delta x[t_k] + B_\Delta u[t_k] \tag{4.9}$$

here $P$ is a matrix that defines the invertible affine mapping between $x[t_k]$ and $\Delta x[t_k]$.

The discrete time algebraic Riccati equation, given by

$$A_\Delta^T S A_\Delta - S - (A_\Delta^T S B_\Delta)(B_\Delta^T S B_\Delta + R)^{-1}(B_\Delta^T S A_\Delta) + Q = 0$$

can now be solved numerically for the matrix $S$. Once the matrix $S$ is calculated, the control gain matrix $K$ for a control of the form $u[x, t_k] = -K\Delta x[t_k] = -K(x[t_k] - z[t_k])$ can be derived for as follows [3]:

$$K = (B_\Delta^T S B_\Delta + R)^{-1}(B_\Delta^T S A_\Delta)$$

The control gain matrix may be calculated recursively at each time step $t_k$ by computer.

### 4.4.1 Asymptotic Tracking Control

The goal of the controller is to track a trajectory based on a periodic input that is known *a priori*. To represent the 'beat' of a musical piece we choose our periodic reference input signal to be

$$z[t_k] = 0.1 \sin 2\pi t_k + 0.1$$

and we define our reference trajectories to be:

$$\begin{aligned} x_1 ref[t_k] &= 0.1 \sin 2\pi t_k + 0.1 \\ x_2 ref[t_k] &= 0.2\pi \cos 0.2\pi t_k \end{aligned} \tag{4.10}$$

We choose these functions because the output of $x_1 ref(t)$ is entirely contained with $[0, \pi]$ (although there is no such restriction on $x_2 ref(t)$) and they are highly differentiable. The system is simulated with the afferent and efferent delays set to $\Delta_a = 0.005$ s and $\Delta_e = 0.035$ s, respectively. The spindle threshold inputs, $\gamma_{fl}$ and $\gamma_{ex}$, are set to zero, while the neural firing inputs, $R_1$ and $R_2$, are determined by the controller. The results of the simulation are depicted in Figure (4.1).

The tracking control algorithm converges asymptotically well for all admissible initial conditions, although in this particular instance they are selected as $x_1[0] = 0.1$ and $x_2[0] = 0$. The propagation delays in the afferent and efferent pathways cause robustness issues, particularly when the arm is changing direction. Such issues result from the delay's exclusion for the state space model. Inclusion of the delays in the model, via Pade or other approximation, is a goal of future research. Inaccuracies in approximating the system with its linearization are manifest in the trajectory's associated muscle forces, plotted in Figure (4.2).

Note also that the differentiability of the muscular forces is lost at several intervals. The sharp transitions reflect the inability of the Jacobian linearization to completely describe the

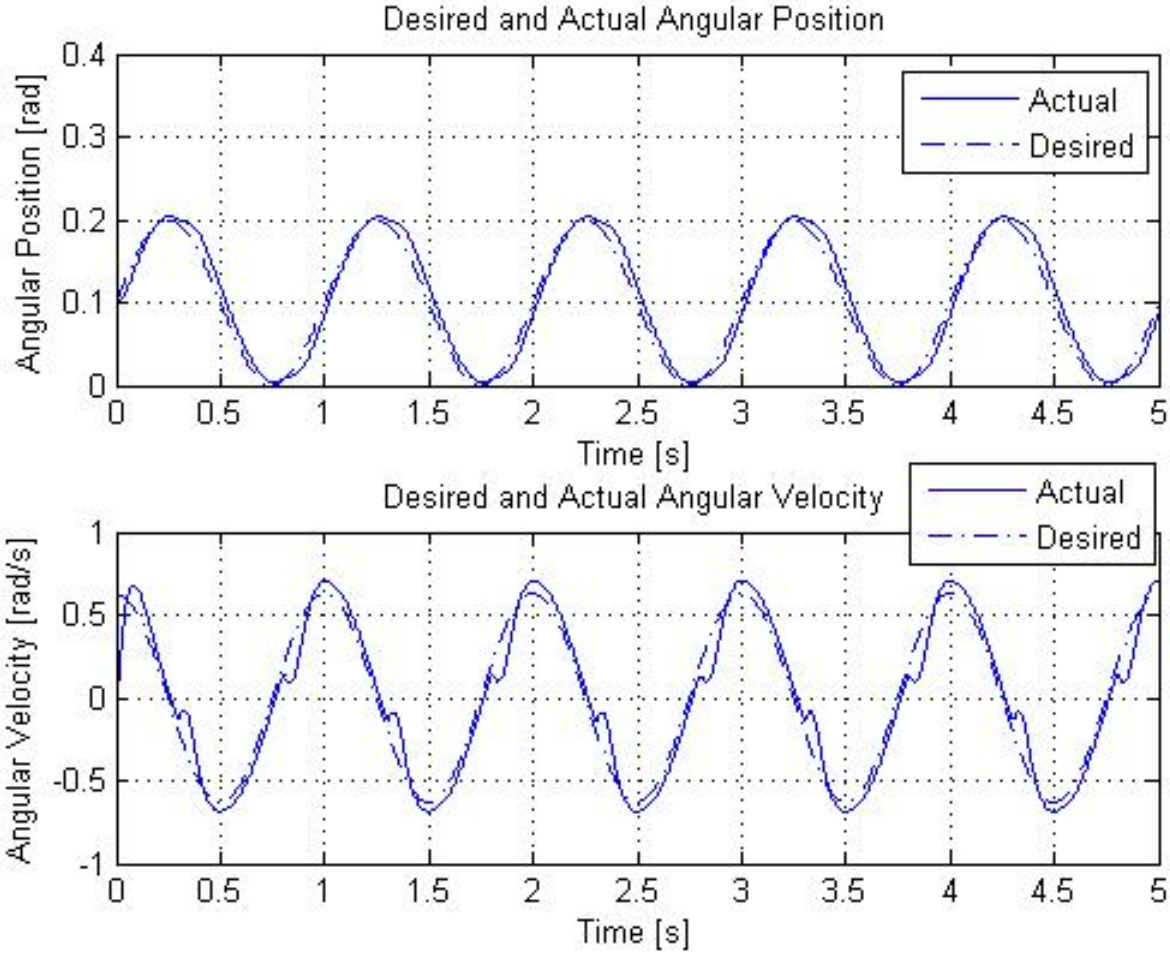


Figure 4.1: State and Reference Trajectories

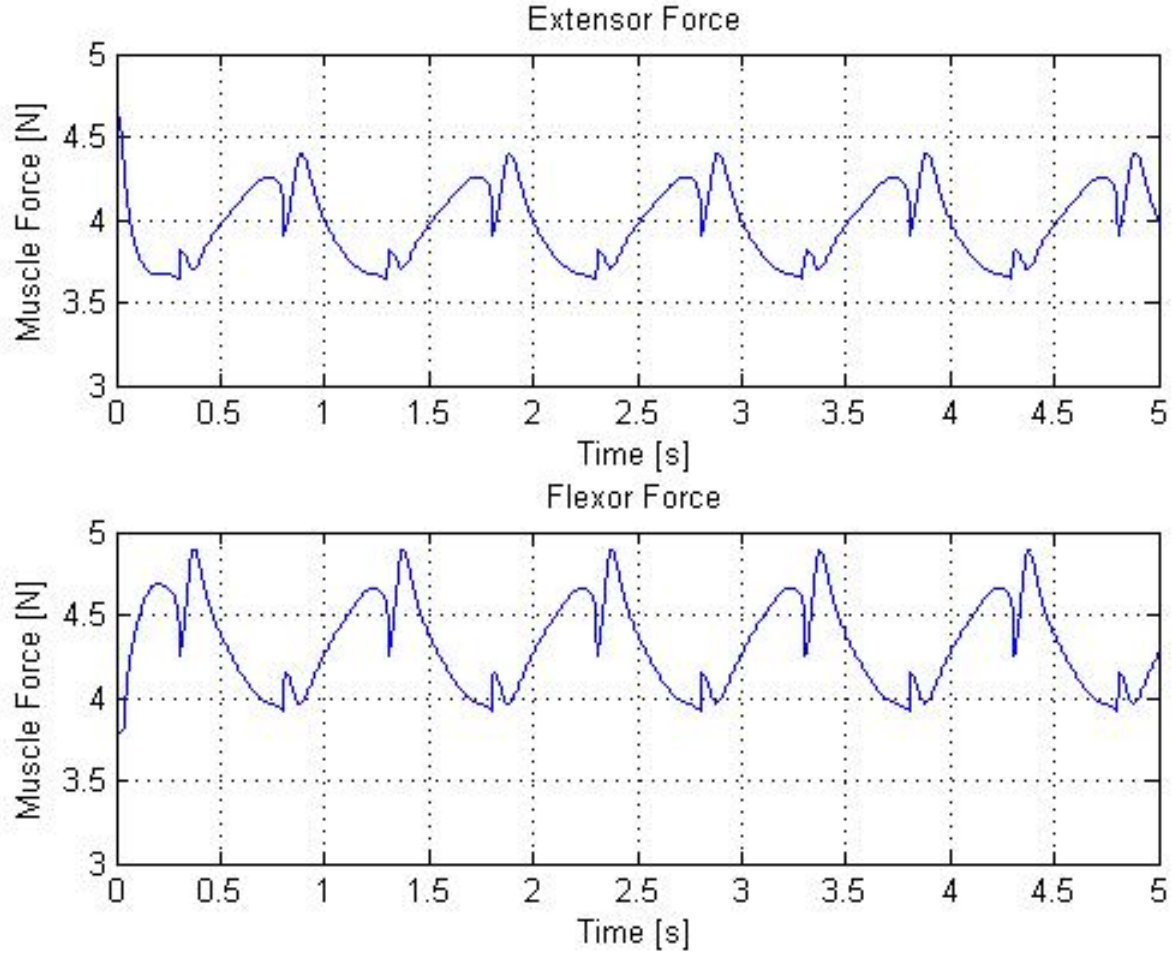


Figure 4.2: Muscle Forces

dynamics of the system. Nevertheless, the control algorithm works fairly well for the class of inputs desired for rhythmic movement. The algorithm will track any smooth reference inputs with fundamental periods much greater than the sampling interval. The control inputs used for this case are plotted in Figure (4.3).

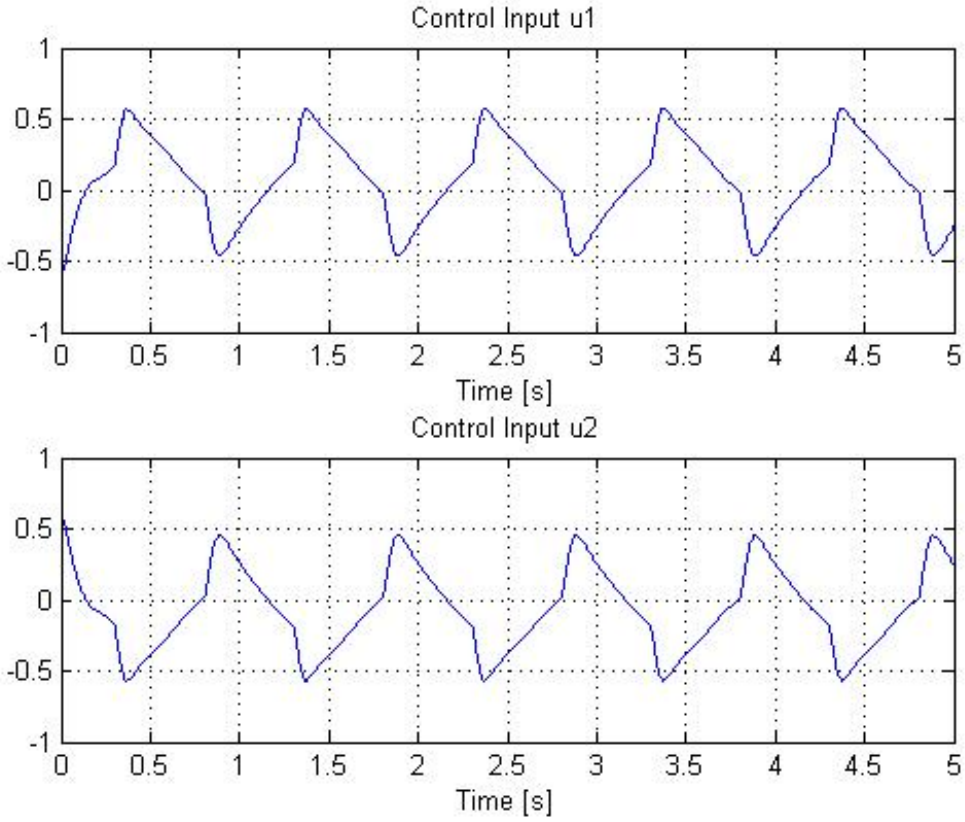


Figure 4.3: Control Inputs

Not surprisingly, the control inputs resemble the respective desired state trajectories. Each of the controls follow the period and extrema of the reference inputs. The control gain matrix elements, depicted if Figure (4.4), are also similar to the reference inputs. Overall, the control algorithm does fairly well in replicating the reference input, even in the face of the significant delays inherent in the system.

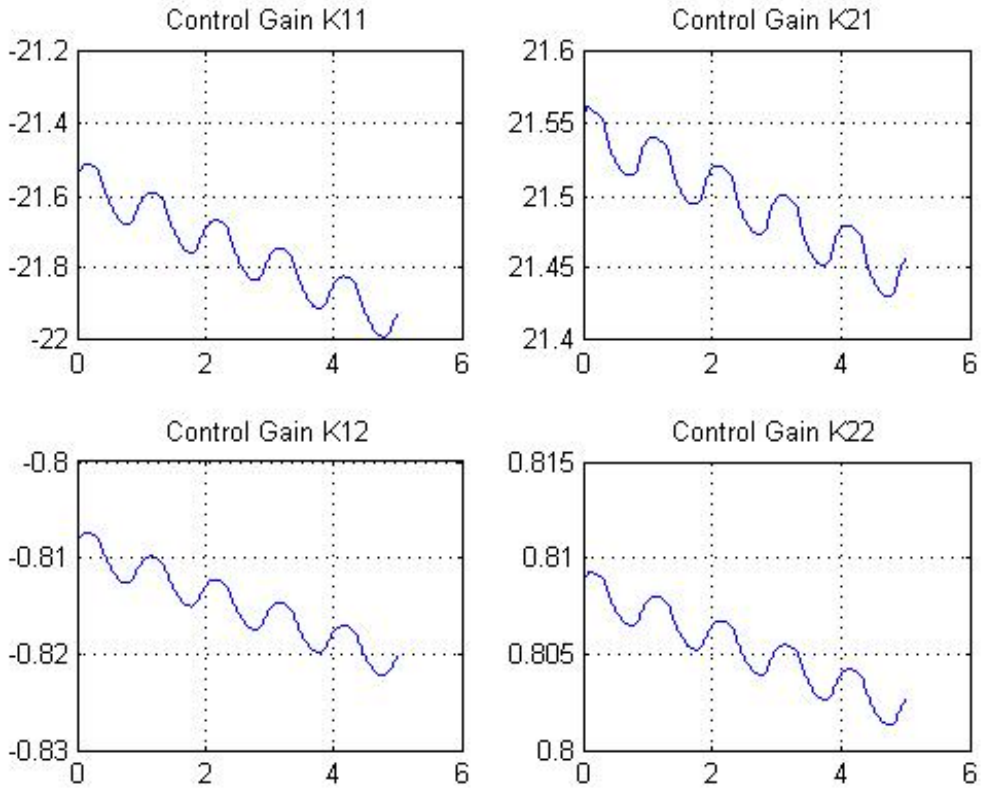


Figure 4.4: Control Gain Matrix Elements

## 4.5 Feedback Linearization of the Second Order Model

The Jacobian linearization used in the previous section presents a number of problems. Although it achieves limited measures of success, the linearization can only drive the system to a close approximation of the desired state trajectory. Additionally, as the model increases in complexity the Jacobian matrix quickly becomes an algebraic mess of nightmarish proportions. We desire a less computationally intensive solution, especially in the design of control laws for more complicated models.

If we consider the system in the coordinates:

$$\begin{aligned} e_1 &= x_1 - z(t) \\ e_2 &= x_2 - \dot{z}(t) \end{aligned}$$

where $z(t)$ is our reference input, the system becomes:

$$\begin{aligned} \dot{e}_1 &= e_2 \\ \dot{e}_2 &= \frac{mgd}{J}\sin x_1 - \frac{k}{J}(4uCkx_1 + Bkx_2) - \ddot{z}(t) \end{aligned} \tag{4.11}$$

The equation (4.11) is in normal form and can be feedback linearized. We desire a control of the form:

$$u(x_1, x_2) = \frac{J}{k^2 4Cx_1}\left(\frac{mgd}{J}\sin x_1 - \frac{k}{J}(Bkx_2) - v(x_1, x_2)\right) \tag{4.12}$$

where the function $v(x_1, x_2)$ is chosen at our discretion. The basic idea of feedback linearization is depicted in the Figure (4.5).

The reduced complexity of the system is an important bonus. The remaining system can be controlled with any of the powerful methods available for linear systems. As a result, we can still employ the optimal control formulation from the previous section, with the added bonus of not having to relinearize the plant at every time step. Thus our control objective can be achieved by judicious choice of $v(x_1, x_2)$. As in the previous section we can derive the

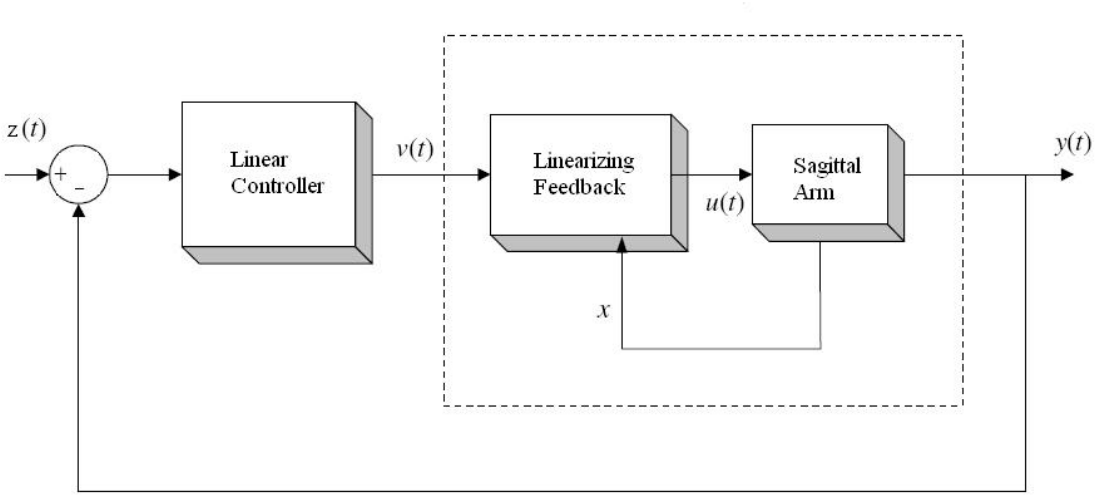


Figure 4.5: The Feedback Linearized System

control $v = -Ke$ from the discrete time Riccati equation. The control gain matrix $K$ need not be calculated online as $A$ and $B$ are now constant matrices.

Unfortunately, a control of the form in equation (4.12) is not well defined because the state $x_1$ (or equivalently: $e_1(t) + z(t)$) is not invertible. Invertibility of $x_1$ is lost because the angular position may be zero at some point over the domain of interest. Thus, a control of this form is not well defined.

## 4.6 Summary

In this chapter two techniques for control of continuous sagittal motion were presented. The first technique was based on Jacobian linearization. Here, the second order model was linearized at various operating points close to the origin. Linearizing the system enables the use of linear feedback control and greatly reduces the system's analytical complexity. Next, the system was sampled at a rate of 1000 Hz to facilitate a computer based solution to the matrix Riccati equation. Finally, a sinusoidal reference input was applied to the linearized discretized system as a tracking goal.

The controlled system was able to achieve accurate angular position and velocity tracking for the sinusoidal reference input. Although some physical inaccuracies were present in algorithm's solution, particularly in the muscle forces, the algorithm tracked fairly well for smooth periodic inputs.

Another promising approach, based on feedback linearization, was explored for the control problem. Unfortunately, a well defined control law could not be derived using this approach. The failure of the strategy was a result of the control being multiplied by a non invertible function of the state in the system equations. Thus an approach using feedback linearization will not be effective for the models presented within this thesis.

# Chapter 5

# Discussion and Conclusions

The goals of this thesis have been reached with a modest amount of success. Several computer models of varying complexity were created within the Simulink environment for use in experimentation. These models, based off of the derivations in Chapter Two, were successfully employed in the study of the arm's dynamics and its associated neural control inputs. The models are far from a true physiological representation of the human arm, and as such all results should be interpreted qualitatively rather than quantitatively.

Stability properties for the second and sixth order model were derived and employed to make deductions regarding the qualitative behavior of system dynamics. Specifically, the golgi feedback organs were found to have a rate increasing effect on the system. Models that incorporate golgi feedbacks will move more quickly in point-to-point movements (in terms or rise time) than those that do not include force and rate of force feedback. Additionally, the spindle feedback signals were found to have a damping effect on the system behavior as they reduced the system settling time and percent overshoot during simulation. The tendon organs themselves were found to be equivalent to low pass filters with force and force derivative feedback. All stability properties that held up for the second order model found analogous counterparts in the sixth order model.

Equally important, analysis of the second order system under complex, continuous movements was also performed. An optimum control problem was formulated to facilitate tracking

of a reference input. The strategy was designed to minimize a cost function based on the arms end point variance, thereby obtaining optimality with respect to both dynamic and kinematic constraints. The resulting sub-optimal strategy successfully tracked a broad class of smooth reference input signals. Specifically, the algorithm was quite successful for sinusoidal signals of varying frequency, amplitude, and phase. Finally, it was noted that although feedback linearization presents a potentially attractive approach, it is not applicable to the current models.

Several future directions exist for further work on this project. Improvement of the sixth order model to more completely encapsulate the properties of the musculotendon actuator pair would be a logical step forward. Alternatively, an analytical model of the system delay would be very valuable in designing a more effective tracking control algorithm. Other strategies might also be investigated to derive control laws that are more computationally efficient, and robust to other classes of reference input. The resolution of these issues could provide the basis for a more rigorous investigation of neural control inputs and muscle forces associated with continuous motion.

# Appendix A

# The Hartman-Grobman Theorem

Let the vector field $f : \Re^N \to \Re^N$ be a class $C^1$ function, with a hyperbolic equilibrium at the origin, that is $f(\underline{0}) = 0$. Then there exists a neighborhood $W \subset \Re^N$ of the origin, and a homeomorphism $h$ (a bijective, continuous function with a continuous inverse) from $W$ to some other neighborhood of $\underline{0}$ with $h \circ f = Af \circ h$. In other words, the system dynamics close to a hyperbolic equilibria are the same, topologically speaking, as those of the linearization [8].

This turns out to be quite useful in our analysis. Essentially, the theorem allows us to approximate the nonlinear system with linear dynamics around an admissible equilibrium point without fear of losing significant dynamical information.

# Appendix B

# Differentiation of the Heaviside Step Function

The Heaviside step function employed in this thesis, defined as:

$$u(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{elsewhere} \end{cases}$$

is not strictly differentiable in the classical sense because it is discontinuous at zero. The loss of differentiability is an extremely undesirable property because it makes the development of a state space version of the model extremely difficult. Additionally, the problems associated with differentiation inhibit the design of control laws based on Jacobian linearizations of the nonlinear plant. We desire a practical solution to this problem. First we will gain insight into the nature of the solution and then we will strengthen our results with a more rigorous mathematical approach.

Suppose instead of the Heaviside step function we consider a function, $f(x)$, which makes a smooth transition (in range) from zero to one. The derivative of this function will be nonzero while $f(x)$ is increasing from zero to one, and zero otherwise. The size of the region where the derivative of our test function is nonzero depends on how long it takes $f(x)$ to increase from zero to one. In the limit that our test function converges to the Heaviside function, the region that the function's derivative is nonzero gets smaller and

smaller. Similarly, the function's transition from zero to one gets steeper and steeper. If the transition period is infinitely short in duration, that is $f(x) = u(x)$, the function's derivative is nonzero over an infinitely small interval where it is infinitely high. A graphical illustration of our converging test function (and its derivative) is provided in Figure (B.1). Note that the area under each derivative curve is identically one by construction.

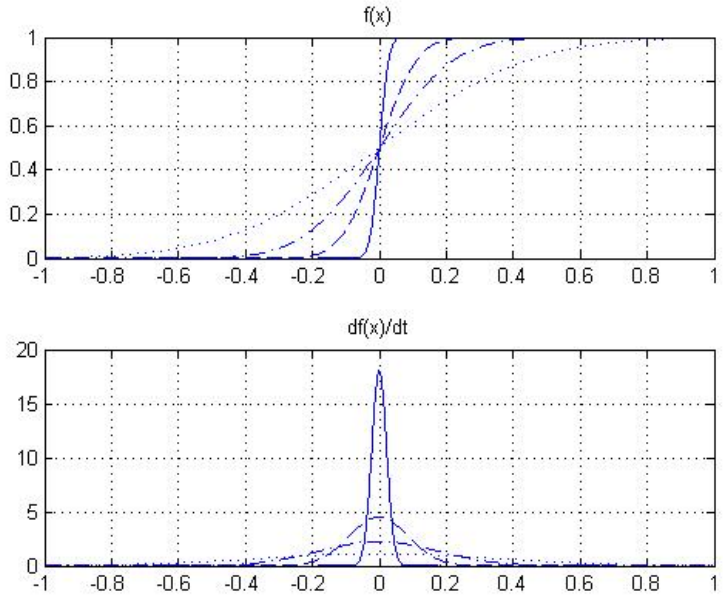


Figure B.1: Differentiating the Heaviside Step: A Graphical View

Now that we have built up some intuition about the properties $du(x)/dx$ should possess, we attempt to give our arguments a mathematically secure foothold. The Heaviside step occurs in our model in products with continuously differentiable functions, so it is this case that we will investigate (although our results are valid in general). Let $g(x)$ be a continuously differentiable function, and suppose we have real numbers $a < 0$ and $b > 0$. Then

$$\begin{aligned} \int_a^b u(x)g'(x)dx &= \int_0^b g'(x)dx \\ &= g(b) - g(0) \end{aligned}$$

by the fundamental theorem of calculus. However, we could have evaluated this integral in another way using integration by parts:

$$\begin{aligned}\int_a^b u(x)g'(x)dx &= [u(x)g(x)]_a^b - \int_a^b u'(x)g(x)dx \\ &= g(b) - \int_a^b u'(x)g(x)dx\end{aligned}$$

If we compare the two results, we obtain

$$\begin{aligned}g(b) - \int_a^b u'(x)g(x)dx &= g(b) - g(0) \\ \int_a^b u'(x)g(x)dx &= g(0)\end{aligned} \tag{B.1}$$

We denote $u'(x) = \delta(x)$. Here $\delta(t)$ is commonly called the Dirac delta function. Although not a function in the strict sense, $\delta(t)$ is typically defined as a distribution or 'generalized' function with the following properties, derived from (B.1),

$$\int_{-\epsilon}^{\epsilon} \delta(t)dt = 1 \qquad \forall \epsilon \in \Re \qquad |\epsilon| > 0$$

$$\delta(t) = 0 \qquad \forall t \in \Re \qquad t \neq 0$$

Note that the Dirac delta is precisely the function we were looking for based on our test function approach (see Figure (B.1)).

For our purposes, any terms that arise in differentiation that contain delta function may be ignored. We are allowed to make this assumption because all of our simulations are performed via computer, in discrete time. Essentially, the delta functions can never be captured by a finite sampling frequency. Such an assumption is intuitively valid for our model because we know all dynamics of the body are well behaved and do not change instantaneously. The mathematical basis for our assumption is derived from the definition of sampling [12]. If we have a continuous signal $x_c(t)$, we may obtain a 'sampled signal' $x_s(t)$ with the following expression.

$$x_s(t) = x_c(t) \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

where $T$ is the sampling interval (e.g. 1 ms).

Note that although, $x_s(t)$ is not a discrete function because its domain is defined over uncountably infinite sets, the above expression captures the essence of the sampling process extremely well. From the definition, we can see immediately that any terms containing delta functions are inevitably lost in the sampling process because they are nonzero over an infinitely small domain.

# Appendix C

# Jacobian Linearization of the Second Order Model

In this section we will derive the Jacobian linearization of the second order model given by equation (2.13):

$$\dot{x_1} = x_2$$

$$\dot{x_2} = \frac{mgd}{J}\sin x_1 + \frac{k}{J}\Big((R_1 - Ckx_1)^2 - Bkx_2 - (R_2 + Ckx_1)^2\Big)$$

where $R_1$ and $R_2$ are given by,

$$\begin{aligned} R_1(x_1, x_2, u_1, u_2) &= \tilde{R_1} q(-kx_2)^{\epsilon}(l_o - kx_1 - \gamma)u(l_0 - kx_1 - \gamma)u(-kx_2) + \tilde{R_1} \\ R_2(x_1, x_2, u_1, u_2) &= \tilde{R_2} q(kx_2)^{\epsilon}(l_o + kx_1 - \gamma)u(l_0 + kx_1 - \gamma)u(kx_2) + \tilde{R_2} \end{aligned}$$

The linearized state space model takes the form:

$$\begin{aligned} \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\Big|_{x=x_{op}, u=u^*} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}\Big|_{x=x_{op}, u=u^*} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\ \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned} \tag{C.1}$$

where $A_{11} = \partial f_1(x)/\partial x_1$, $A_{12} = \partial f_1(x)/\partial x_2$, $A_{21} = \partial f_2(x)/\partial x_1$, $A_{22} = \partial f_2(x)/\partial x_2$, $B_{11} = \partial f_1(x)/\partial u_1$, $B_{12} = \partial f_2(x)/\partial u_1$, $B_{21} = \partial f_2(x)/\partial u_1$, and $B_{22} = \partial f_2(x)/\partial u_2$ respectively. The Jacobian linearization with respect to the state is given by,

$$A_{11} = 1$$

$$A_{12} = 0$$

$$\begin{aligned}
A_{21} = \frac{mgd\cos(x_1)}{J} + \frac{2k}{J}\Bigg(&\Big(u_1 q k^{\epsilon}(-x_2)^{\epsilon}(l - kx_1 - \gamma)u(l - kx_1 - \gamma)u(-kx_2) + u_1 + Cl - Ckx_1\Big) \\
&\cdot\Big( - u_1 q k^{\epsilon}(-x_2)^{\epsilon} k u(l - kx_1 - \gamma)u(-kx_2) - u_1 q k^{\epsilon}(-x_2)^{\epsilon}(l - kx_1 - \gamma)\delta(l - kx_1 - \gamma)ku(-kx_2) \\
&-Ck\Big) - \Big(u_2 q k^{\epsilon} x_2^{\epsilon}(l + kx_1 - \gamma)u(l + kx_1 - \gamma)u(kx_2) + u_2 + Cl + Ckx_1\Big) \\
&\cdot\Big(u_2 q k^{\epsilon} x_2^{\epsilon} k u(l + kx_1 - \gamma)u(kx_2) + u_2 q k^{\epsilon} x_2^{\epsilon}(l + kx_1 - \gamma)\delta(l + kx_1 - \gamma)ku(kx_2) + Ck\Big)\Bigg)
\end{aligned}$$

$$\begin{aligned}
A_{22} = \frac{k}{J}\Bigg(&2\Big(u_1 q k^{\epsilon}(-x_2)^{\epsilon}(l - kx_1 - \gamma)u(l - kx_1 - \gamma)u(-kx_2) + u_1 + Cl - Ckx_1\Big) \\
&\cdot\Big(\frac{u_1 q k^{\epsilon}(-x_2)^{\epsilon}\epsilon(l - kx_1 - \gamma)u(l - kx_1 - \gamma)u(-kx_2)}{x_2} - u_1 q k^{\epsilon}(-x_2)^{\epsilon}(l - kx_1 - \gamma)u(l - kx_1 - \gamma) \\
&\cdot\delta(kx_2)k\Big) - Bk - 2\Big(u_2 q k^{\epsilon} x_2^{\epsilon}(l + kx_1 - \gamma)u(l + kx_1 - \gamma)u(kx_2) + u_2 + Cl + Ckx_1\Big) \\
&\cdot\Big(\frac{u_2 q k^{\epsilon} x_2^{\epsilon}\epsilon(l + kx_1 - \gamma)u(l + kx_1 - \gamma)u(kx_2)}{x_2} + u_2 q k^{\epsilon} x_2^{\epsilon}(l + kx_1 - \gamma)u(l + kx_1 - \gamma)\delta(kx_2)k\Big)\Bigg)
\end{aligned}$$

and the Jacobian with respect to the control is,

$$B_{11} = 0$$

$$B_{12} = 0$$

$$\begin{aligned}
B_{21} = \frac{2k}{J}\Bigg(&u_1\, q k^{\epsilon}(-x_2)^{\epsilon}\,(l - kx_1 - \gamma)\,u\,(l - kx_1 - \gamma)\,u\,(-kx_2) + u_1 + Cl - Ckx_1\Big) \\
&\cdot\Big(q k^{\epsilon}(-x_2)^{\epsilon}\,(l - kx_1 - \gamma)\,u\,(l - kx_1 - \gamma)\,u\,(-kx_2) + 1\Big)
\end{aligned}$$

$$B_{22} = \frac{-2k}{J}\Big(u_2\, q k^{\epsilon} x_2{}^{\epsilon}\,(l + kx_1 - \gamma)\,u\,(l + kx_1 - \gamma)\,u\,(kx_2) + u_2 + Cl + Ckx_1\Big)$$

$$\cdot\Big(qk^{\epsilon}{x_2}^{\epsilon}\left(l+kx_1-\gamma\right)u\left(l+kx_1-\gamma\right)u\left(kx_2\right)+1\Big)$$

We ignore any terms in the linearization containing Dirac delta functions because such terms are nonzero over an infinitely small interval, and thus can never be captured by a finite sampling frequency. Such an assumption is intuitively valid for our model because we know all dynamics of the body are well behaved and do not change instantaneously. Accordingly, the following simplifications to the linearized continuous time system are made:

$$\begin{aligned}
A_{21} = {} & \frac{mgd\cos(x_1)}{J}+\frac{2k}{J}\Bigg(\Big(u_1qk^{\epsilon}(-x_2)^{\epsilon}(l-kx_1-\gamma)u(l-kx_1-\gamma)u(-kx_2)+u_1+Cl-Ckx_1\Big)\\
& \cdot\Big(-u_1qk^{\epsilon}(-x_2)^{\epsilon}ku(l-kx_1-\gamma)u(-kx_2)-Ck\Big)-\Big(u_2qk^{\epsilon}x_2^{\epsilon}(l+kx_1-\gamma)u(l+kx_1-\gamma)u(kx_2)\\
& +u_2+Cl+Ckx_1\Big)\cdot\Big(u_2qk^{\epsilon}x_2^{\epsilon}ku(l+kx_1-\gamma)u(kx_2)Ck\Big)\Bigg)\\
A_{22} = {} & \frac{k}{J}\Bigg(2\Big(u_1qk^{\epsilon}(-x_2)^{\epsilon}(l-kx_1-\gamma)u(l-kx_1-\gamma)u(-kx_2)+u_1+Cl-Ckx_1\Big)\\
& \cdot\Big(\frac{u_1qk^{\epsilon}(-x_2)^{\epsilon}\epsilon(l-kx_1-\gamma)u(l-kx_1-\gamma)u(-kx_2)}{x_2}\Big)-Bk-2\Big(u_2qk^{\epsilon}x_2^{\epsilon}(l+kx_1-\gamma)\\
& \cdot u(l+kx_1-\gamma)u(kx_2)+u_2+Cl+Ckx_1\Big)\Big(\frac{u_2qk^{\epsilon}x_2^{\epsilon}\epsilon(l+kx_1-\gamma)u(l+kx_1-\gamma)u(kx_2)}{x_2}\Big)\Bigg)
\end{aligned}$$

Mathematically speaking, the linearization is indeterminate at $x_2 = 0$, however, we assign any undefined terms a value of zero in such instances. Such action is in keeping with the intent of the spindle sensor model, where the feedback signal vanishes at $x_2 = 0$ as a result of Heaviside step functions.

# Appendix D

# Matlab Functions

```
%%Project:      A Biological Control Model of a One Link Sagittal Arm
%%Author:       Michael Lucas
%%Last Update:  2/10/06
%%Known Bugs:   None
%%Comments: Simulink Block Diagrams follow easily from the
%% equations presented in Chapter Two.

function [sys,x0,str,ts]=sagittal_arm_sfc(t,x,u,flag,X1_init,X2_init)

switch flag

    case 0  %initialization

        str=[];
        ts=[0 0];
        s=simsizes

            s.NumContStates=2;
            s.NumDiscStates=0;
            s.NumOutputs=2;
            s.NumInputs=1;
            s.DirFeedthrough=0;
            s.NumSampleTimes=1;

        sys=simsizes(s);
        x0= [X1_init,X2_init];

    case 1  %derivatives

        F=u;
```

```
        sys=sagittal_arm(t,x,F);

    case 3  %Output

        sys=x;

    case {2 4 9}    % 2: Discrete
                    % 4: CalcTimeHit
                    % 9: Termination
        sys=[];

    otherwise

        error(['Unhandled Flag =',num2str(flag)]);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function dx = sagittal_arm(t,x,F)

%Model for the sagittal arm
%**************************************************************************
%The differential equations are:
%
%dx_1/dt=x_2
%
%dx_2/dt=(1/J)*(m*g*d*sin(x_1)+f)

MGD=1.9*10*0.18; %Mass of arm * gravity constant * distance of the center
                 %of gravity from the base [Kg*m^2/s^2]

J=0.075;     %moment of inertia about the base in [Kg.m^2], unloaded arm
             %J=0.165 moment of inertia of loaded arm relative to the base

X_1=x(1);    %Extracting the Angular Position [rad]
X_2=x(2);    %Extracting the Angular Velocity {rad/s]

X1_dot=X_2;
X2_dot=(1/J)*(MGD*sin(X_1)+F);

dx=[X1_dot,X2_dot];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [sys,x0,str,ts]=DTNS_sfc(t,x,u,flag,X1_init,X2_init,U1_init,U2_init)

switch flag

    case 0  %initialization

        str=[];
        ts=[0 0];
        s=simsizes

            s.NumContStates=4;
            s.NumDiscStates=0;
            s.NumOutputs=4;
            s.NumInputs=4;
            s.DirFeedthrough=0;
            s.NumSampleTimes=1;

        sys=simsizes(s);
        x0= [X1_init,X2_init,U1_init,U2_init];

    case 1  %Derivatives

        x=u;
        sys=x;

    case 3  %Output


        sys=DTNS(t,x,u);

    case {2 4 9}    % 2: Discrete
                    % 4: CalcTimeHit
                    % 9: Termination
        sys=[];

    otherwise

        error(['Unhandled Flag =',num2str(flag)]);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function K = DTNS(t,x,u)

%Discrete Linear Model for the sagittal arm
%*********************************************************************

MGD=1.9*10*0.18; %Mass of arm * gravity constant * distance of the center
                 %of gravity from the base [Kg*m^2/s^2]
J=0.075;         %moment of inertia about the base in [Kg.m^2]
C=30;            %muscle Information constant
B=75;            %muscle Information constant
l_0=0;           %length of arm [m]
k=0.4;
T=0.001;         %sampling period [s]

X_1=x(1);        %Extracting the Angular Position [rad]
X_2=x(2);        %Extracting the Angular Velocity [rad/s]
U_1=x(3);        %Extracting the Neural Control Input R1
U_2=x(4);        %Extracting the Neural Control Input R2

A=[0,1;MGD/J*cos(X_1)+2*k^2*C/J*(U_1+U_2+2*C*l_0),k^2*B/J];
B2=[0,0;-2*k/J*(U_1+C*l_0-C*k*X_1),2*k/J*(U_2+C*l_0-C*k*X_1)];

A=eye(2)+T*A;
B2=T*B2;

Q=[1 0;0 1];
R=eye(2);

[gain,S,e] = dlqr(A,B2,Q,R);

K=[gain(1,1);gain(1,2);gain(2,1);gain(2,2)];
```

# Bibliography


[1] Dabney, James B. and Harman, Thomas L. *Mastering Simulink*; Prentice Hall. Upper Saddle River, New Jersey; 2004.

[2] Dinneen, Janet A. and Hemami, Hooshang *Stability and Movement of a One-Link Neuromusculoskeletal Sagittal Arm*; IEEE Transactions on Biomedical Engineering, Vol. 40:6, pp 541-548; 1993.

[3] Gorczyca, Przemyslaw and Hajduk, Krystyn *Tracking Control Algorithms for a Laboratory Aerodynamical System*; Int. J. Appl. Math. Comput. Sci., Vol. 14:4, pp 469-475; 2004

[4] Kawato, Mitsuo *Internal Models for Motor Control and Trajectory Planning*; Current Opinion in Neurobiology, Vol. 9, pp 718-727; 1999.

[5] Khalil, Hassan K. *Nonlinear Systems*; Prentice Hall. Upper Saddle River, New Jersey; 2002.

[6] Kim, Jaywoo and Hemami, Hooshang *Control of a One-Link Arm by Burst Signal Generators*; Biological Cybernetics, Vol. 73, pp 37-47; 1995.

[7] Mendel, Jerry M. *Lessons in Estimation Theory for Signal Processing, Communcations, and Control*; Prentice Hall. Upper Saddle River, New Jersey; 1995.

[8] Rayskin, Victoria; Guysinisky, Misha; and Hasselblatt, Boris. *Differentiability of the Hartman-Grobman Linearization*; Discrete and Continuous Dynamical Systems, Vol. 9:4, pp. 979-984; 2003

[9] Pinch, Enid R *Optimal Control and the Calculus of Variations* Oxford University Press. Oxford; 1993.

[10] Rugh, Wilson J. *Linear System Theory*; Prentice Hall. Upper Saddle River, New Jersey; 1996.

[11] Simmons, Gavin and Demiris, Yiannis *Optimal Robot Arm Control Using the Minimum Variance Model*; Journal of Robotic Systems, Vol. 22:11, pp 677-690; 2005.

[12] Vetterli, M. and Kovacevic, J. *Wavelets and Subband Coding*; Prentice Hall. Upper Saddle River, New Jersey; 1995.

[13] Yamaguchi, G. T. and Zajac, Felix E. *Restoring Unassisted Natural Gait to Paraplegics Via Functional Neuromuscular Stimulation: A Computer Simulation Study*; IEEE Transactions on Biomedical Engineering, Vol. 37:9, pp 886-903; 1990.

[14] Zajac, Felix E. and Levine, William S. *Novel Experimental and Theoretical Approaches To Study the Neural Control of Locomotion and Jumping*; Posture and Movement, pp 259-279. Edited by R. E. Talbott and D. R. Humphrey. Raven Press, New York; 1979.

[15] Zajac, Felix E. *Muscle and Tendon: Properties, Models, Scaling, and Application to Biomechanics and Motor Control*; Critical Reviews in Biomedical Engineering, Vol. 17:4, pp 359-411.; 1989